ABSTRACT

Title of Dissertation: ADAPTIVE SUPERPOSITION OF FINITE ELEMENT MESHES IN LINEAR AND NONLINEAR DYNAMIC ANALYSIS

Zhihua Yue, Doctor of Philosophy, 2005

Directed By: Assistant Professor Donald H. Robbins, Jr.
Department of Mechanical Engineering

The numerical analysis of transient phenomena in solids, for instance, wave propagation and structural dynamics, is a very important and active area of study in engineering. Despite the current evolutionary state of modern computer hardware, practical analysis of large scale, nonlinear transient problems requires the use of adaptive methods where computational resources are locally allocated according to the interpolation requirements of the solution form. Adaptive analysis of transient problems involves obtaining solutions at many different time steps, each of which requires a sequence of adaptive meshes. Therefore, the execution speed of the adaptive algorithm is of paramount importance. In addition, transient problems require that the solution must be passed from one adaptive mesh to the next adaptive mesh with a bare minimum of solution-transfer error since this form of error compromises the initial conditions used for the next time step.

A new adaptive finite element procedure (s-adaptive) is developed in this study for modeling transient phenomena in both linear elastic solids and nonlinear elastic solids caused by progressive damage. The adaptive procedure automatically updates the time step size and the spatial mesh discretization in transient analysis, achieving the accuracy and the efficiency requirements simultaneously. The novel feature of the s-adaptive procedure is the original use of finite element mesh superposition to produce spatial refinement in transient problems. The use of mesh superposition enables the s-adaptive procedure to completely avoid the need for cumbersome multipoint constraint algorithms and mesh generators, which makes the s-adaptive procedure extremely fast. Moreover, the use of mesh superposition enables the s-adaptive procedure to minimize the solution-transfer error.

In a series of different solid mechanics problem types including 2-D and 3-D linear elastic quasi-static problems, 2-D material nonlinear quasi-static problems, and 2-D transient problems for linear elastic and material nonlinear materials, the s-adaptive solution is compared to a solution obtained using a non-adaptive, uniform refined mesh. These comparisons clearly demonstrate that the s-adaptive method is capable of generating a solution with the same accuracy level as a non-adaptive, uniform refined mesh; however, the s-adaptive solution uses far fewer DOF and consequently executes much faster.

ADAPTIVE SUPERPOSITION OF FINITE ELEMENT MESHES IN LINEAR AND NONLINEAR DYNAMIC ANALYSIS

By

Zhihua Yue

Dissertation submitted to the Faculty of the Graduate School of the
University of Maryland, College Park, in partial fulfillment
of the requirements for the degree of
Doctor of Philosophy
2005

Advisory Committee:
Assistant Professor Donald Robbins, Jr., Chair/Advisor
Professor Abhijit Dasgupta
Professor Balakumar Balachandran
Professor Donald Barker
Professor Sung Lee

Dedication

To my grandparents who passed away during my Ph.D. study.

Acknowledgements

I would like to express my most sincerely gratitude to my advisor, Professor Donald H. Robbins, Jr., for his guidance and encouragement in the course of the present study. This dissertation would not have been possible without his continuous support. He was always there for me with his outstanding knowledge, keen insights and brilliant inspirations. He was always willing to sacrifice his precious time to accommodate me. Over the past four years, Professor Robbins has shown many of the best academic qualities and personal characteristics that I admire and respect deeply. I was very fortunate and honored to have him as my advisor.

I am grateful to Professors Sung Lee, Abhijit Dasgupta, Donald Barker, Balakumar Balachandran, Amr Baz and Hugh Bruck for taking time from their extremely demanding schedules to serve in my dissertation committee. Their time and efforts involved in evaluating my dissertation and providing the valuable suggestions are greatly appreciated.

I am grateful to Dr. Karin Robbins for her kind and generous assistance in helping me settle down smoothly in the United States

I appreciate the precious friendship between my friends and me. With listening ears and helping hands, my friends bring tons of joy to my life.

There are no words to thank my parents enough for their love and blessing. They have done so much for me. I wish them healthy and happy. My gratitude also goes to my sister who shouldered the burden of taking care of our parents solely when I was in pursuit of graduate study.

Finally, I am thankful to my lovely fiancée, Lan. She shows me a whole new world and fills my life with sunshine and joy. My love and gratitude towards her are endless.

Table of Contents

List of Tables

List of Figures

Chapter 1: INTRODUCTION AND BACKGROUND

1.1 General Remarks

The analysis of transient phenomena in elastic solids, for instance, wave propagation and structural dynamics, is an important area of study in modern civil, mechanical and aerospace engineering. We are interested in how stress waves propagate in solids and how structures behave under dynamic or time-varying loads. An offshore drilling platform under the strike of waves, a high-speed train running on a railway and an airplane in flight are some typical examples of transient phenomena in elastic solids.

Ever since the first landmark paper by Clough [1] in early 1960s, the finite element method (FEM), has been flourishing with the rapid and vast developments in both hardware and software of the digital computer. Tremendous research efforts have been and are being devoted to this methodology in laying down solid theoretical foundations, developing robust and efficient algorithms, implementing those algorithms on computers and extending the methodology to virtually every field of applied science and engineering. It has been successfully applied in an extremely wide range of complex scientific and engineering problems. The development history of the FEM may be found in review papers by Babuska [2] and Zienkiewicz [3].

The FEM is a powerful and indispensable tool for the analysis of engineering problems that can be modeled mathematically by partial differential equations. An infinite DOF system in transient analysis is governed by partial differential equations with respect to both the space and the time coordinates. Naturally, FEM has been widely applied in transient analysis. In essence, FEM is an approximate technique that uses an assembly of discrete elements to approximate a continuum. The FEM divides the domain of the problem under consideration into a finite number of simple subdomains, i.e., the finite elements, and then uses variational/weak formulations to construct a piecewise polynomial approximation of the solution. The FEM aims at obtaining an approximate numerical solution of an actual physical problem whose real solution is rarely known in general. Therefore, several types of errors are inevitably present in the numerical solution obtained by the FEM. These errors include the modeling errors due to the simplifications inherent in the chosen mathematical model, round-off errors due to the finite digit arithmetic performed on computers, and the discretization errors that are caused by approximating the entire solution domain by a set of finite elements, and then assuming a simple solution form over each individual element. In this study, we shall only be concerned with the discretization errors, which in practice, may be intuitively defined as the difference between the exact analytical solution of the governing partial differential equations and the computed FEM solution.

To ensure the accuracy of the computed FEM approximate solution, it is essential to properly measure (or estimate) the discretization errors. Two commonly used error estimates are *a priori* error estimates and *a posteriori* error estimates. *A priori* error estimates show the finite element solutions converge to the corresponding exact solutions to the original partial differential equations under certain conditions. However, as exact solutions are generally unavailable, the majority of the efforts on estimating errors are devoted to *a posteriori* error estimates which assess discretization errors by utilizing the computed finite element solutions. To solve a general engineering problem with a user-specified accuracy requirement, in general, the analyst does not know what finite element discretization is sufficient enough in advance. In this case, the analysis may be carried out on a coarse mesh first. If *a posteriori* error estimator indicates the solution has not met the accuracy requirement, it is desirable to somehow refine the existing FEM model, most likely, by refining the mesh, to efficiently seek a new solution which meets the accuracy requirement. This is the basic concept of the *adaptive finite element method*.

The overall goal of an adaptive finite element solution is to achieve a specified level of solution accuracy with the least overall computational effort. In practical terms, the adaptive solution process involves iteratively determining an efficient finite element mesh that is capable of delivering a solution that meets the stated accuracy requirements, while at the same time keeping the number of degrees of freedom to a minimum in the final chosen mesh. The cost of iteratively determining such a mesh contributes to the

overall computational cost of the solution; therefore, it is imperative that the adaptivity algorithms should be fast and efficient.

It has been well proven that adaptive finite element analysis is very powerful in reducing the computation time and memory requirements when solving partial differential equations whose solution exhibits localized behavior in the space domain. Compared with the conventional FEM, the adaptive FEM introduces more intelligence into the analysis. The two key issues in the adaptive FEM are the estimation of discretization errors and the refinement of the FE models. Obviously, the refinement technique plays an essential role in adaptive FEM. However, the significance of error estimates should not be neglected. *A posteriori* error estimators not only provide information about the accuracy of the solution, but also shed light on the distributions and intensities of discretization errors such that FE models will have a chance to be refined accordingly.

Since the pioneering work of Babuska and Rheinboldt [4] in the mid 1970s, many *a posteriori* error estimators have been developed for *linear elliptic* problems (for example, static stress analysis problems and steady-state heat conduction problems). Basically, the widely used error estimators can be categorized into two major types [5]: the residual type error estimators [4] and the recovery type error estimators [6]. The residual type error estimators measure the errors by examining the residuals of the finite element approximation arising in the element equilibrium equations and the element tractions

across the inter-element and the external boundaries. The recovery type error estimators, or Zienkiewicz-Zhu (Z^2) error estimators, assess the errors from a more intuitive point of view by using a more accurate recovered (i.e., postprocessed) solution to replace the generally unknown exact solution in calculating errors. Obviously, its effectiveness strongly relies on the accuracy of the recovered solution obtained from the postprocessing procedure. In 1992, Zienkiewicz and Zhu [7, 8] developed the famous *Superconvergent Patch Recovery* (SPR) technique, which exploits the superconvergent gradients (stresses in most cases) at certain sampling points in finite element meshes. By the least square fitting of a proper polynomial expansion to the superconvergent stresses in the local patch surrounding, the SPR procedure is able to recover a continuous stress field with the same accuracy as the stresses at the sampling points, and eventually recovers superconvergent stresses field over the whole domain. Zienkiewicz et al. [9] claimed that recovery type error estimation has many advantages over residual type error estimation, such as conceptual simplicity, superior accuracy and less computational intensity. Wiberg and Abdulwahab [10-12] further enhanced the SPR technique by taking into account the additional constraints such as the residual of equilibrium and the imbalance in boundary conditions.

Different techniques are available for FE mesh refinement. Basically, the refinement techniques fall into the following categories [13]: a) *h-refinement* changes the mesh, generally by reducing the element sizes but still using the same class of elements; b)

p-refinement increases, generally hierarchically, the polynomial order of interpolation functions but keeping the mesh unchanged; and c) *r-refinement* relocates the nodes toward critical area while keeping the number of nodes in mesh unchanged. On occasions it is possible to combine the above refinement schemes to form new schemes, for example, *hp-refinement* or *hr-refinement*.

The h-refinement is the most popular refinement technique due to its simplicity and easy implementation. Basically, the h-refinement can be divided into two subclasses [5]: 1) *element subdivision/enrichment* where refinement is conveniently achieved by dividing the existing elements into smaller elements while keeping the original element boundaries intact. 2) *mesh regeneration* or *remeshing* where the old mesh is destroyed and the new mesh is generated based on the new element sizes predicted for the whole mesh. Both refining and coarsening can be simultaneously performed.

Despite of the abundant existence of 2-D mesh generators and the claimed possibility of extending their 2-D mesh generators to 3-D by many researchers, 3-D mesh generators are scarce primarily due to the extreme geometric complexity triggered by the additional spatial dimension. Since 3-D models are so commonplace these days, automated mesh refinement schemes for 3-D problems are more desirable than ever. The simplest h-refinement is practically the most effective scheme for 3-D problems due to its simplicity. Although the remeshing type h-refinement scheme is widely used in 2-D

adaptive analysis, it becomes impractical for general 3-D use mainly due to the programming difficulty and computational cost of repeatedly regenerating the entire 3-D finite element mesh. Moreover, to implement this scheme, a complicated and powerful mesh generator is necessary to produce a new mesh based on the required element size distribution and it is very difficult to obtain such mesh generator for certain types of elements [5]. Also, the complete remeshing scheme poses difficulty in transferring solution data between the old mesh and the new mesh. Therefore, researchers resort to the element subdivision/enrichment type h-refinement scheme. However, hanging nodes may be created where an element with a mid-side node is connected to an element without such node. Special transition elements [14] or multi-point constraints still have to be introduced, which may reduce the efficiency.

The abovementioned difficulties are overcome in a different type of h-refinement—the *hierarchical h-refinement*, also termed as *s-refinement* or *mesh superposition technique*. It was firstly proposed by Fish [15] who superimposed independent local overlay meshes on specific subregions of a pre-existing global mesh to produce a refined composite mesh for 2-D elastostatic problems. The displacement field that is interpolated on the local overlay meshes serves as an additive enhancement (or relative displacement field) to the displacement field that is interpolated on the original global mesh. Due to the hierarchical nature of the composite displacement field, the overlay mesh and global mesh need not have compatible discretizations; therefore, the overlay mesh can be superimposed

anywhere on the global mesh to provide interpolation enhancement precisely where it is desired, regardless of the geometry of the original mesh. This technique was applied to the analysis of composite laminates [16-20] and was combined with homogenization theory to carry out the multi-scale computation in composites [21]. A similar methodology, termed as the *Nested FEM*, was applied in the stress analysis of the electronic products [22, 23]. Two-level mesh superposition was extended to multi-level adaptive mesh superposition (s-adaptive) and applied to various types of linear elastostatic problems in the literatures [24-26].

To date, tremendous progresses have been accomplished on adaptive finite element methods for elliptic problems, such as elastostaticity and heat conduction problems [27-29]. The adaptive analysis of elliptic problems is relatively simple because the error comes solely from spatial discretization, and error control can be effectively achieved by an adaptive remeshing alone. However, the development of adaptive finite element methods for hyperbolic problems such as dynamic problems is far from complete, yet remains highly desirable.

In elastodynamic problems, regions of high stress and strain gradients do not have fixed sizes, shapes or positions, but evolve over time. This is especially true in the case of stress wave propagation where the rapidly moving regions of high stress gradients typically exhibit a high degree of localization at any instant in time. While it is

theoretically possible to solve such problems with a uniform refined mesh, this approach is impractical due to the fact that the mesh over the entire computational domain must be refined to a level where the wave can be accurately resolved, regardless of its instantaneous location. Instead, it is desirable to periodically adjust the spatial mesh and time step size based on the evolving characteristics of the solution, to ensure that at any point in time, the mesh and time step size are sufficiently refined in the regions and stages with high stress gradients and reasonably coarse in others.

Thus far, the predominant formulation for dynamic problems is the semidiscrete finite element method [30-32] where the spatial domain is first discretized into finite element mesh, resulting in a system of second-order ordinary differential equations in time, which are also called structural dynamic equations. Subsequently, the ordinary differential equations are converted to a sequence of linear algebraic systems via various direct numerical time integration methods. Obviously, two kinds of discretization errors are inherently rooted in the semidiscrete finite element method: 1) the spatial discretizaion error resulting from the finite element approximation; 2) the temporal discretization error resulting from the time integration method. Therefore, unlike in the elliptic problems, to develop adaptive approach for dynamic problems, it is necessary to develop *a posteriori* error estimators for both spatial and temporal errors which can then be used to adaptively adjust the level of both spatial and temporal discretization in an attempt to control these errors.

While this adaptive approach in dynamics is simple in concept, two major difficulties are encountered in its actual implementation for large-scale dynamic problems. First, the repeated adjustments that must be made to the mesh add significantly to the overall computational expense, not to mention the fact that mesh adjustment is itself a rather complex task. The computation expense is further increased by the need to compute local error estimators which are used to guide the adaptive changes to the mesh and time step size. Second, each time the mesh is altered, the solution from the previous mesh must be interpolated onto the new mesh before the solution can proceed. In most cases, this interpolation process is not exact due to differences in the two meshes, thus introducing additional error into the solution. For transient analyses, this additional error is quite significant since it represents error in the initial conditions for the next time step.

While many works have been conducted on adaptive methods for dynamic problems, the vast majority of these works focus only on spatial adaptivity or temporal adaptivity, thus the adaptive methods are not complete since both spatial errors and temporal errors should be dealt with in dynamic analysis.

Among the works on spatial adaptivity, many *a posteriori* error estimators for elliptic problems have been extended to dynamic problems. Joo and Wilson [33] solved the structural dynamic problems by finite elements in space and load dependent Ritz vector superposition in time. Mesh refinements were carried out based on the residual type of

error estimator. Cook and Avrashi [34] discussed the error estimation and adaptive meshing for vibration problems. Belyschko and Tabbara [35] studied h-adaptive procedures for transient solid mechanics problems with emphasis on localizations due to material instability. After various error criteria were tested, a strain type Zienkiewicz-Zhu error estimator was proposed and found most effective for such dynamic shear band problems. Based on the Zienkiewicz-Zhu error estimator and a mesh generator, Zeng and Wiberg [36, 37] presented a spatial mesh adaptation procedure which used an automatic remeshing scheme and was capable of monitoring the movement of steep stress gradient regions by updating the spatial mesh according to a given error tolerance. Their studies demonstrated the successful extension of the Zienkiewicz-Zhu error estimate to many dynamic problems. However, time discretization errors have been ignored in all of these studies.

Among those works on temporal adaptivity, there are few associated with the commonly used time integration methods, such as the Newmark method [38] and HHT-α method [39]. Generally, when a single-step method like the Newmark method is used, local temporal errors, which are defined as the errors resulting from a single time step, are first estimated and can be used to adaptively control the time step size. Bergan and Mollestad [40] suggested an objective criterion for the performance and guideline for making an adaptive time stepping algorithm for practical applications. Zienkiewicz and Xie [41] presented a simple local error estimator and an adaptive time-stepping procedure for the

Newmark method by comparing the Newmark solutions with the exact solutions expanded in Taylor series. Zeng *et al.* [42] obtained the same result in a more intuitive way. Wiberg and Li [43] derived a formulation for linearly varied third-order derivatives and thus developed a more precise error estimator to access the errors of both displacement and velocity, and furthermore, error in total energy norm, by a postprocessing technique. Choi and Chung [44] proposed a quadratic function for the approximation of the locally exact value of the acceleration and obtained the local error simply by comparing the time integration solutions with the locally exact solutions.

There are a few published papers simultaneously considering the effects of space and time discretization for transient analysis. Wiberg and Li [45] developed a remeshing type of h-version adaptive finite element procedure which is able to update the spatial mesh and time step automatically so that both the spatial and time discretization errors are under control. However, their method seems not capable of conserving the total energy even for an undamped system. Besides introducing interpolation errors to the solution, their solution interpolation process based on the remeshing approach is also time consuming. This problem is compounded by the fact that the combination of a suitable mesh and suitable time step are iteratively determined during each mesh change. The new mesh is not optimal for the following analysis because it is observed that the adaptive procedure may update the spatial mesh more frequently than necessary. Choi and Chung [46] proposed an adaptive procedure for 1-D problems which consistently estimated and

controlled the spatial and temporal discretization errors. However, the remapping error that occurs in control procedure of spatial discretization error becomes considerably large for problems without damping. Therefore, more extended studies were needed on remapping techniques. Furthermore, the use of an effective automatic mesh generator is necessary for the method to be extended to 2-D and 3-D problems.

In summary, existing methods for mesh refinement and solution interpolation rely predominantly on brute force, i.e. complete remeshing and complete re-interpolation of the solution. This is conceptually simple, but poses three major shortcomings:

1) In general, these brute force methods are not fast and efficient enough for large-scale transient analysis. Even though computer hardware has evolved considerably in the past decade, this is no excuse for the use of inefficient, brute force methods. Improvements in algorithm efficiently will allow currently available computers to solve larger problems with faster turn-around times.

2) Complete remeshing requires the subsequent complete re-interpolation of the solution which adds a new source of numerical error into the solution. This 're-interpolation error' effectively contaminates the initial conditions that are used for integrating over each time step that follows a mesh change.

3) While the brute force methods have been successfully applied to moderately sized 2-D transient problems, they quickly become impractical to use for 3-D transient problems. This is primarily due to the programming difficultly and computational expense associated with the repeated regeneration of a 3-D finite element mesh. Methods are needed that can be extended from 2-D to 3-D without a disproportionate increase in programming complexity and solution expense.

With the adaptive procedures successfully applied in the linear transient analysis of structures, the nonlinear transient analysis of structures seems to be a particularly promising field for adaptive analysis. First, it is the most computationally demanding application among various types of structural applications due to the potential iterations needed to account for the nonlinearity. Furthermore, a priori selection of an appropriately refined mesh is most difficult, if not totally impossible, since the optimal mesh configuration changes continually throughout the generally unpredictable deformation process. At last, for the nonlinear transient problems with high localization of deformation, high strains usually occur in a very small part of the domain, for example, the most severe deformation in a buckled thin wall beam may be localized over 5% of the surface area [47], which makes it practically impossible to conduct analysis on a uniform refined mesh and thus adaptive analysis is almost a necessity. The advantage of the adaptive methods is that they place the degrees of freedom where and only where they are needed, so that the most accurate solution can be achieved for a given amount of

computational resources. Noting that the need for high computational efficiency is ever increasing in modern society, the considerable gain in efficiency by adaptive analysis is highly desirable.

Along with the great prospects, the adaptive analysis of the nonlinear transient problem is also the most challenging. Despite the considerable maturity and wide application of adaptive procedures in linear elliptic problems, due to the inherent complexity, relevant studies on non-elliptic nonlinear problems such as dynamic plasticity or progressive damage are still in the stages of development. Meanwhile, it is particularly desirable to accurately and yet efficiently solve the non-elliptic nonlinear problems as they play more and more important role in modern science and engineering. Although it is natural and reasonable to extend an adaptive methodology that was initially developed for linear problems to nonlinear problems, the extension may not be as straightforward as it appears. Many difficulties are involved in developing an adaptive methodology for the nonlinear problems and much research work has been devoted to the subject [35, 47-63]. These difficulties include the development of suitable error estimators, data transfer schemes and mesh refinement schemes.

Two main categories can be identified in the literature for error estimators in nonlinear adaptivity [48]: the first based on gradients or curvature of displacement, which is generally used in localization problems [49, 50] and can only indicate the existence of

error but not its magnitude; and the second based on postprocessing/recovery, which has exhibited excellent performance [9] in linear analysis and is adopted in this study.

Unlike the path-independent solutions of linear problems, the solutions of the materially nonlinear problems are generally path-dependent, which means the solutions are related to the history of deformations. Accordingly, the loading is applied in an incremental manner with an iterative linearization, using procedures such as Newton-Raphson method in each incremental step. In the incremental formulation of the nonlinear finite element method, there are four basic state variables in each increment, namely, total stresses, total strains, stress increments and strain increments, available to estimate the errors. Various error estimators have been proposed in the literature using various combinations of the four basic variables, depending on the characteristics of the problem under consideration and also the objective of the adaptive analysis.

The use of the total stresses (or strains) in L_2 norm yields the L_2 norm of stress (or strain) error, which is simple and easy to compute and thus has been extensively used in both geometrically and materially nonlinear problems [35, 52, 64]. Belytschko and Tabbara [35] used L_2 norm of strain error to solve transient problems involving localization due to material instability. Wiberg and Li [54] applied the same error estimator to nonlinear structural dynamic analysis. For most elastoplastic problems it has been shown that this kind of error estimator is not quite effective [55, 56]. The use of the stress (or strain)

increment in L_2 norm yields the L_2 norm of incremental stress (or strain) error. The use of both the total stresses and the strain increments in L_2 norm yields so-called energy rate error norm [55]. This error estimator is an improvement to L_2 norm of stress/strain error because it resolves the error due to the total stresses and the error due to the strain increments. Note that this error estimator necessitates the calculation of two improved solutions at each increment step: one being the improved total stresses, and the other being the improved strain increments. Peric et al. [55] used this type of error estimator where the improved fields of stresses and strains were computed by L_2 norm of Zienkiewicz [6]. Gallimard et al. [56] defined an error measure in the constitutive relations based on Drucker's inequality for elastoplastic problems. The recovery achieves a stress field which satisfies the equilibrium equation in each increment. The recovered strain rate field is achieved by using the recovered stress field in the constitutive relations. Boroomand and Zienkiewicz [48] also adopted this type of error estimator where the improved stresses were computed by the SPR (Supercovergent Patch Recovery) procedure and the improved strain increments were computed by REP (Recovery by Equilibrium in Patches). It was the first time the powerful recovery procedure such as SPR and REP were applied in nonlinear problems of plasticity [48].

Considering the path-dependency of materially nonlinear problems, Li and Bettess [51] developed a true path-dependent error estimator which is related not only to the current incremental calculation but also to all previous incremental calculations. The absolute

error in each increment is calculated in the energy norm of the strain increment. The total error in the total solution is treated as a superposition of the errors in each past increment. Physically, the error estimator represents an accumulation measure of the errors in all past increment steps.

There are two kinds of state variables in finite element analysis: *node-wise* (internal) variables, namely, the quantities stored at nodes of finite element mesh (i.e. displacements, velocities); and *element-wise* variables, namely, the quantities stored at integration points of finite elements (i.e. plastic deformations, hardening). When a remeshing is performed in adaptive analysis, a *transfer operator* is used to transfer the state variables from the old mesh to the new mesh. It is obvious that the transferring of the *node-wise* variables is mandatory for the solution process to proceed and it is commonly accomplished by a simple direct interpolation from the old nodes to the new ones using the interpolation functions on the old mesh [47, 48, 54, 57]. Though the transferring of the *element-wise* variables is not generally necessary in linear problems, it is mandatory in materially nonlinear problems because these history-dependent *element-wise* variables involved in the nonlinear constitutive equations in general cannot be recomputed from the initial state. The transferring of *element-wise* variables is quite challenging and has been addressed by many investigators [48, 49, 57-59].

To date, the majority of the research on transferring of *element-wise* variables has focused on the strategy of "project→transfer→interpolate". First, the element-wise variables are projected to the nodes of the old mesh and then further transferred to the nodes of the new mesh where they are temporarily treated as node-wise variables. Next, these temporary node-wise variables are used to interpolate the corresponding node-wise values on the new mesh. Finally, the resulting node-wise variables of the new mesh are used to interpolate the appropriate element-wise variables at the element integration points in the new mesh. Different methods can be distinguished based primarily on the method chosen for projecting the element-wise variables to the nodes. Lee and Bathe [59] adopted a least square scheme in a similar manner as the SPR procedure using a patch of elements. Peric et al. [57] proposed a patch-wise computation of nodal values using the average of extrapolated fields from the integration points.

Peric et al. [57] pointed out several important aspects of the transfer operation, such as consistency with the constitutive equations, requirements of equilibrium, compatibility of the transferred displacement field, compatibility with evolving boundary conditions and minimization of the numerical diffusion of the state fields. Boroomand and Zienkiewicz [48] pointed out the need for self-consistency of the history-dependent parameters based on the fact that the commonly used procedure of "project→transfer→interpolate" violates the equilibrium of the system even when there is no mesh change involved, which can be easily verified by performing the procedure in a single mesh while the material is elastic.

In other words, after mesh adaptation and then data transfer, it is not guaranteed or possible, to be more specific, that the transferred state variables on the new mesh satisfy both the consistency condition and the equilibrium condition. Therefore, the selection of one of these above conditions usually is made based on the engineering judgement. The satisfaction of the constitutive consistency condition and the neglect of the equilibrium condition are sometimes preferred [57, 59]. As the third and the forth requirements of compatibility (Peric [57]) are relatively easier to satisfy, the research is more focused on the fifth requirement, namely, minimization of the numerical diffusion of the state fields. It is known that the element-wise variables of the old mesh are first projected to nodes before being transferred. In a global L_2 projection, the projection is conducted over the whole problem domain and thus the local information will be spread over a comparatively larger zone. In contrast, the spread of local information is much smaller for patch-wise approaches such as SPR procedure. The SPR procedure was used by Zienkiewicz and Boroomand [60] to obtain nodal stresses. Unfortunately, the process is not always convergent. The same authors [48] proposed a *Direct Gauss point to Gauss point* transfer operator which uses weighted form of least-square procedure in the same sense as the recovery procedure. The main advantage of this method is that it prevents the spread of the local information over a large area when several remeshing is required in the adaptive analysis [48]. Belytschko et al. [47] proposed a *fission and fusion* adaptive remeshing procedure relating to the explicit dynamic analysis of shells. The strategy consists of splitting a single parent quadrilateral element into four equal-sized sibling

quadrilateral elements (fission) when the parent element needs refinement, or combing four sibling elements into their parent element (fusion) when the refinement is no longer necessary for the sibling elements. In the fission process, the newly generated four sibling elements inherit the element-wise variables from their parent element. In the fusion process, the element-wise variables for the fused parent element are taken to be the area-weighted average of the original sibling elements variables. In this way, the material history diffusion is minimized and transfer of the state variables is not a serious issue [57].

1.2 Aims and Outline

The aim of the present study is to develop an adaptive finite element modeling methodology that is fast and efficient enough to solve large-scale linear and nonlinear transient problems. The computational speed and efficiency that are demanded by this class of problems is obtained by extending the finite element mesh superposition method to transient problems. The emphasis of the research is on the development of a general solution methodology for transient problems, not the study of a particular transient solid mechanics phenomenon. To this end, an s-version adaptive (*s-adaptive*) procedure is developed for the semidiscrete finite element analysis of dynamic problems. This s-adaptive procedure is able to automatically update the spatial mesh and the time step so

that both temporal and spatial discretization errors are controlled within user-specified tolerances and the computational cost is nearly minimal.

The s-adaptive procedure overcomes the three major weaknesses of the earlier adaptive efforts [45, 46]. The three major shortcomings are the lack of computational efficiency, the unnecessary introduction of interpolation errors and the impracticality of extending the methodology from 1-D or 2-D to 3-D analysis.

With this chapter as a prelude, the dissertation consists of eight chapters. An outline of the remainder is given as follows.

In Chapter 2, the basic semidiscrete finite element method which uses the HHT-α method for time integration is reviewed for 2-D linear dynamic problems. The sources of discretization errors are identified and *a posteriori* error estimates are discussed. In time, the technique in [43, 45] is used for local temporal error estimate. In space, the superconvergence patch recovery (SPR) technique proposed by Zienkiewicz-Zhu [7, 8] is used to obtain spatial error.

In Chapter 3, the formulation and the characteristics of the mesh superposition technique [15] with emphasis on the computer implementation are presented.

Chapter 4 is devoted to the spatial error estimator used in the present study. The basic formulation of the superconvergent patch recovery (SPR) technique is first reviewed. A rank deficiency problem in the SPR procedure with 4-node quadrilateral elements is detected and reported. A *rotated SPR* procedure is thus developed to simply and efficiently solve the rank deficiency problem by utilizing a local rotated coordinate system. Finally, a new *Layer-by-Layer SPR* scheme is developed to utilize the SPR technique in the composite meshes generated by the mesh superposition technique.

In Chapter 5, the s-adaptive method, which uses the mesh superposition technique to provide mesh refinement, is presented in detail and its performance has been demonstrated in 2-D linear static problem. Subsequently, the s-adaptive method is extended to more complex 3-D linear elastic problems. The extension is straightforward and do not require a disproportionate increase in algorithm development difficulty or solution time. The numerical examples demonstrate the superior characteristic of the s-adaptive method in 3-D problems.

In Chapter 6, an s-adaptive procedure for semidiscrete finite element method is developed for 2-D linear elastic transient problems. The performance characteristics of the s-adaptive procedure is also studied and evaluated. Numerical examples are presented to show the high performances of the proposed s-adaptive procedure over the traditional h-adaptive procedure in terms of efficiency and accuracy.

In Chapter 7, the proposed s-adaptive procedure is extended to the analysis of 2-D material nonlinear transient problems whose nonlinearity is caused by progressive damage. First, the fundamental background of progressive damage is reviewed. And then, the difficulty of calculating system matrices and vectors in the composite mesh which is posed by the spatially varying material properties is addressed. Moreover, a *mastergrid* data transfer scheme is developed to ensure that the computed distribution of internal (or element-wise) variables is accurately transferred between consecutive composite meshes. Finally, numerical examples are presented to show the high performances of the proposed s-adaptive procedure.

Finally in Chapter 8, the conclusions are drawn and some future developments are recommended.

1.3 Innovative Features

Some of the material presented in this dissertation have been published previously or will be published soon in [65, 66]. Some of ideas developed in this dissertation that are believed to be novel at the time of publication are:

A *Layer-by-Layer SPR* scheme is developed to utilize the standard SPR technique in the composite meshes generated by the mesh superposition technique.

The rank deficiency problem in the SPR procedure with 4-node quadrilateral elements is detected and reported. A *rotated SPR* procedure is developed to simply and efficiently solve the rank deficiency problem by utilizing a local rotated coordinate system.

The extension of the s-adaptive method to 3-D linear static problems is implemented.

An s-adaptive procedure which is able to automatically update spatial mesh and time step is developed for 2-D linear elastic transient problems.

An s-adaptive procedure is developed for the analysis of 2-D material nonlinear transient problems caused by progressive damage.

Chapter 2: ERRORS AND THEIR ESTIMATIONS IN THE SEMIDISCRETE FINITE ELEMENT FORMULATION

In this chapter, the basic semidiscrete finite element formulations for linear dynamic problems are reviewed and the sources of discretization errors are identified in section 2.1. *A posteriori* error estimates are discussed in section 2.2, where the approach by Wiberg and Li [43, 45] is used for local error estimate in time domain and the superconvergent patch recovery technique by Zienkiewicz and Zhu [7, 8] is used to obtain spatial error in space domain.

2.1 Errors in the Semidiscrete Finite Element Formulation

In the semidiscrete finite element formulation, the spatial variation of the dependent variable (displacement field) is first approximated over a finite element mesh as $\mathbf{u}(\mathbf{x},t) \approx \mathbf{N}(\mathbf{x})\mathbf{d}(t)$, where $\mathbf{N}(\mathbf{x})$ is a matrix of known global interpolation functions and $\mathbf{d}(t)$ is an unknown vector of time dependent displacement degrees of freedom. Substitution of this spatial approximation into the virtual work expression yields a simultaneous 2nd order system of ordinary differential equations in time, which can be written in matrix form as

$$\mathbf{M}\ddot{\mathbf{d}}(t)+\mathbf{C}\dot{\mathbf{d}}(t)+\mathbf{K}\mathbf{d}(t)=\mathbf{F}(t) \quad \textit{for } 0<t<T, \quad \text{subjected to } \mathbf{d}(0)=\mathbf{d}_0, \quad \dot{\mathbf{d}}(0)=\dot{\mathbf{d}}_0 \qquad (2.1)$$

where **M** is the mass matrix, **C** is the damping matrix, **K** is the stiffness matrix, **F** is the load vector, and $\mathbf{d}$, $\dot{\mathbf{d}}$, $\ddot{\mathbf{d}}$ are the unknown nodal displacement, velocity and acceleration vectors, respectively, and $\mathbf{d}_0$ and $\dot{\mathbf{d}}_0$ represent known initial conditions. Letting $\mathbf{d}(t)$ denote the exact solution to equation (2.1), and letting $\mathbf{u}(\mathbf{x},t)$ denote the exact solution to the original non-discretized dynamic problem, in spatial approximation we actually have

$$\mathbf{u}(\mathbf{x},t) \approx \underline{\mathbf{u}}^{\mathrm{h}}(\mathbf{x},t) = \mathbf{N}(\mathbf{x})\mathbf{d}(t) \tag{2.2}$$

where $\underline{\mathbf{u}}^{\mathrm{h}}(\mathbf{x},t)$ denotes a finite element approximation of the exact solution $\mathbf{u}(\mathbf{x},t)$ to the transient problem that is void of any temporal error (i.e., contains only spatial error). In this case, the spatial error $\mathbf{e}_{\mathrm{s}}$ is expressed as

$$\mathbf{e}_{\mathrm{s}} \equiv \mathbf{u} - \underline{\mathbf{u}}^{\mathrm{h}} \tag{2.3}$$

In practice, equation (2.1) is not solved exactly; instead, we obtain an approximate solution to equation (2.1) via direct numerical integration which requires further discretization with respect to the time domain. This is the discretization in time domain. Here, we adopt the HHT-α variant [39] of the Newmark direct integration method which improves the Newmark method and consists of the following recursive equations:

$$\mathbf{M}\ddot{\mathbf{d}}_{\mathrm{n+1}} + (1+\alpha)\mathbf{C}\dot{\mathbf{d}}_{\mathrm{n+1}} - \alpha\mathbf{C}\dot{\mathbf{d}}_{\mathrm{n}} + (1+\alpha)\mathbf{K}\mathbf{d}_{\mathrm{n+1}} - \alpha\mathbf{K}\mathbf{d}_{\mathrm{n}} = \mathbf{F}(t_{\mathrm{n+1+\alpha}}) \tag{2.4a}$$

$$\dot{\mathbf{d}}_{\mathrm{n+1}} = \dot{\mathbf{d}}_{\mathrm{n}} + [(1-\gamma)\ddot{\mathbf{d}}_{\mathrm{n}} + \gamma\ddot{\mathbf{d}}_{\mathrm{n+1}}]\Delta t_{\mathrm{n}} \tag{2.4b}$$

$$\mathbf{d}_{n+1} = \mathbf{d}_n + \dot{\mathbf{d}}_n \Delta t + [(1-2\beta)\ddot{\mathbf{d}}_n + 2\beta\ddot{\mathbf{d}}_{n+1}]\frac{\Delta t_n{}^2}{2} \qquad (2.4c)$$

where $\mathbf{d}_n$, $\dot{\mathbf{d}}_n$, $\ddot{\mathbf{d}}_n$ are the discrete computed approximations of the exact solution $\mathbf{d}(t_n)$, $\dot{\mathbf{d}}(t_n)$, $\ddot{\mathbf{d}}(t_n)$ of equations (2.1). In equations (2.4b,c), $\Delta t_n = t_{n+1} - t_n$ is the time step size at time t_n, and $t_{n+1+\alpha} = t_{n+1} + \alpha\Delta t_n$. The values of the parameters α, γ and β control the stability and accuracy characteristics of HHT-α method. If the parameters are chosen such that $-1/3 \le \alpha \le 0$, $\gamma = (1-2\alpha)/2$ and $\beta = (1-\alpha)^2/4$, then the method is unconditionally stable with 2nd order accuracy. The HHT-α method reduces to the Newmark method when α=0. Equation (2.4a) represents the modified time discrete equations of motion in terms of approximate solutions $\mathbf{d}_n$, $\dot{\mathbf{d}}_n$, $\ddot{\mathbf{d}}_n$, and equation (2.4b,c) define the evolution of the approximate solutions. The approximate solution of equation (2.1) via equations (2.4) introduces further error which can be characterized as temporal discretization error $\mathbf{e}_t$, defined as the difference between $\mathbf{d}_n$ and $\mathbf{d}(t_n)$.

After the spatial and temporal discretizations, the final raw finite element solution is

$$\mathbf{u}^h(\mathbf{x},t_n) = \mathbf{N}(\mathbf{x})\mathbf{d}_n, \quad \dot{\mathbf{u}}^h(\mathbf{x},t_n) = \mathbf{N}(\mathbf{x})\dot{\mathbf{d}}_n, \quad \boldsymbol{\sigma}^h(\mathbf{x},t_n) = \mathbf{DSd}_n \qquad (2.5a,b,c)$$

where **D** is the constitutive matrix, and **S** is the strain/displacement matrix. The *total* errors for the displacements, velocities and stresses in the semidiscrete finite element solution can are expressed as

$$\mathbf{e} = \mathbf{u} - \mathbf{u}^h, \qquad \dot{\mathbf{e}} = \dot{\mathbf{u}} - \dot{\mathbf{u}}^h, \qquad \mathbf{e}_\sigma = \boldsymbol{\sigma} - \boldsymbol{\sigma}^h \qquad (2.6a,b,c)$$

where $\mathbf{e}$, $\dot{\mathbf{e}}$ and $\mathbf{e}_\sigma$ contain both spatial and temporal errors.

For consistent measurement of the temporal and spatial errors in a dynamic analysis, we will consider the impact of distributed error on the total energy *E*, which is expressed as the sum of the total kinetic energy and the total strain energy, and can be written as

$$E(\mathbf{u},\dot{\mathbf{u}}) = \frac{1}{2}\left(\int_\Omega \dot{\mathbf{u}}^{\mathrm{T}} \rho \dot{\mathbf{u}} d\Omega + \int_\Omega \boldsymbol{\sigma}^{\mathrm{T}} \mathbf{D}^{-1} \boldsymbol{\sigma} d\Omega \right) \tag{2.7}$$

Accordingly, the error in the total energy norm can then be written as

$$\left\| \mathbf{e} \right\|_{\mathrm{E}} = \left\| \mathbf{u} - \mathbf{u}^{\mathrm{h}} \right\|_{\mathrm{E}} \equiv \left(2E(\mathbf{e},\dot{\mathbf{e}}) \right)^{1/2} = \left[\int_\Omega \dot{\mathbf{e}}^{\mathrm{T}} \rho \dot{\mathbf{e}} d\Omega + \int_\Omega \mathbf{e}_\sigma{}^{\mathrm{T}} \mathbf{D}^{-1} \mathbf{e}_\sigma d\Omega \right]^{1/2} \tag{2.8}$$

2.2 Estimations of the Errors by Postprocessing

In practical applications, the exact solution $\mathbf{u}$ is generally not available. To estimate the error, Zienkiewicz and Zhu [6] proposed an effective postprocessing/recovery method for linear elliptic problems. This idea then has been extensively and effectively used in practical applications. By postprocessing the original finite element solutions $\mathbf{u}^{\mathrm{h}}$, $\dot{\mathbf{u}}^{\mathrm{h}}$ and $\boldsymbol{\sigma}^{\mathrm{h}}$, higher order accurate solutions $\mathbf{u}^{*}$, $\dot{\mathbf{u}}^{*}$ and $\boldsymbol{\sigma}^{*}$ can be recovered. These recovered solutions then serve the function of the unknown exact solutions in the estimation of the errors. The differences between the recovered solutions and the finite element solutions are used to estimate the errors.

For dynamic problems, the total error in the energy norm can be estimated as

$$\left\|\mathbf{e}\right\|_{\mathrm{E}} \cong \left\|\underline{\mathbf{e}}\right\|_{\mathrm{E}} = \left\|\mathbf{u}^{*} - \mathbf{u}^{\mathrm{h}}\right\|_{\mathrm{E}} = \left[\int_{\Omega} (\dot{\mathbf{u}}^{*} - \dot{\mathbf{u}}^{\mathrm{h}})^{\mathrm{T}} \rho (\dot{\mathbf{u}}^{*} - \dot{\mathbf{u}}^{\mathrm{h}}) d\Omega + \int_{\Omega} (\boldsymbol{\sigma}^{*} - \underline{\boldsymbol{\sigma}}^{\mathrm{h}})^{\mathrm{T}} \mathbf{D}^{-1} (\boldsymbol{\sigma}^{*} - \underline{\boldsymbol{\sigma}}^{\mathrm{h}}) d\Omega\right]^{1/2} \quad (2.9)$$

However, equation (2.9) contains contributions from both spatial and temporal error. It is desirable to estimate the temporal errors and spatial errors separately in order to independently assess the adequacy of the time step size and spatial discretization. For this reason, the error in equation (2.6a) is rewritten as

$$\mathbf{e} = \mathbf{u} - \mathbf{u}^{\mathrm{h}} = (\mathbf{u} - \underline{\mathbf{u}}^{\mathrm{h}}) + (\underline{\mathbf{u}}^{\mathrm{h}} - \mathbf{u}^{\mathrm{h}}) = \mathbf{e}_{\mathrm{s}} + \mathbf{e}_{\mathrm{t}} \quad (2.10)$$

Recall that $\underline{\mathbf{u}}^{\mathrm{h}}$ represents the approximate solution that includes only spatial errors. In general, $\underline{\mathbf{u}}^{\mathrm{h}}$ is unknown. According to [67], we know that for any choice of norm, the following relationship holds,

$$\left\|\mathbf{e}\right\|_{\mathrm{E}} = \left\|\mathbf{e}_{\mathrm{s}} + \mathbf{e}_{\mathrm{t}}\right\|_{\mathrm{E}} \le \left\|\mathbf{e}_{\mathrm{s}}\right\|_{\mathrm{E}} + \left\|\mathbf{e}_{\mathrm{t}}\right\|_{\mathrm{E}} \quad (2.11)$$

Therefore, the temporal errors and spatial errors can be estimated separately and the upper bound of total discretization error in norm can be obtained by the sum of the two errors. In this study, we use this upper bound to approximate the total error. That is

$$\left\|\mathbf{e}\right\|_{\mathrm{E}} \approx \left\|\mathbf{e}_{\mathrm{s}}\right\|_{\mathrm{E}} + \left\|\mathbf{e}_{\mathrm{t}}\right\|_{\mathrm{E}} \quad (2.12)$$

where the contributions of the spatial and temporal discretization errors to the error in the total energy norm can be expressed as

$$\left\|\mathbf{e}_s\right\|_E = \left\|\mathbf{u} - \underline{\mathbf{u}}^h\right\|_E = \left[\int_\Omega (\dot{\mathbf{u}} - \underline{\dot{\mathbf{u}}}^h)^T \rho (\dot{\mathbf{u}} - \underline{\dot{\mathbf{u}}}^h) d\Omega + \int_\Omega (\boldsymbol{\sigma} - \underline{\boldsymbol{\sigma}}^h)^T \mathbf{D}^{-1} (\boldsymbol{\sigma} - \underline{\boldsymbol{\sigma}}^h) d\Omega\right]^{1/2} \quad (2.13a)$$

$$\left\|\mathbf{e}_t\right\|_E = \left\|\underline{\mathbf{u}}^h - \mathbf{u}^h\right\|_E = \left[\int_\Omega (\underline{\dot{\mathbf{u}}}^h - \dot{\mathbf{u}}^h)^T \rho (\underline{\dot{\mathbf{u}}}^h - \dot{\mathbf{u}}^h) d\Omega + \int_\Omega (\underline{\boldsymbol{\sigma}}^h - \boldsymbol{\sigma}^h)^T \mathbf{D}^{-1} (\underline{\boldsymbol{\sigma}}^h - \boldsymbol{\sigma}^h) d\Omega\right]^{1/2} \quad (2.13b)$$

2.2.1 Postprocessing in Time

Given the finite element solution $\mathbf{u}^h$, the higher order accurate solution $\underline{\mathbf{u}}^*$ is obtained from the improved solutions $\mathbf{d}^*$, $\dot{\mathbf{d}}^*$ that are computed during the temporal postprocessing of $\mathbf{d}_n$, $\dot{\mathbf{d}}_n$ (i.e. the solution of equation (2.4)). In this study, we adopt the strategy by Wiberg [42, 43] where improved solutions $\mathbf{d}^*$ and $\dot{\mathbf{d}}^*$ are computed based on the assumption of 3rd order time derivatives of the displacements that vary linearly over each time step. *A posteriori* error estimation of displacement and velocity are then evaluated as the differences between $\mathbf{d}^*, \dot{\mathbf{d}}^*$ and $\mathbf{d}_n$, $\dot{\mathbf{d}}_n$. After $\mathbf{d}^*$ and $\dot{\mathbf{d}}^*$ are obtained, $\underline{\boldsymbol{\sigma}}^h$ and $\underline{\dot{\mathbf{u}}}^h$ in equations (2.13) can be approximated as

$$\underline{\boldsymbol{\sigma}}^h \approx \underline{\boldsymbol{\sigma}}^* = \mathbf{S}\mathbf{N}\mathbf{d}^* \qquad \text{and} \qquad \underline{\dot{\mathbf{u}}}^h \approx \underline{\dot{\mathbf{u}}}^* = \mathbf{N}\dot{\mathbf{d}}^* \quad (2.14a,b)$$

Furthermore, the time discretization error $\left\|\mathbf{e}_t\right\|_E$ in equation (2.12) is approximated by $\left\|\underline{\mathbf{e}}_t\right\|_E$ as

$$\left\|\mathbf{e}_t\right\|_E \approx \left\|\underline{\mathbf{e}}_t\right\|_E = \left\|\underline{\mathbf{u}}^* - \mathbf{u}^h\right\|_E = \left[\int_\Omega (\underline{\dot{\mathbf{u}}}^* - \dot{\mathbf{u}}^h)^T \rho (\underline{\dot{\mathbf{u}}}^* - \dot{\mathbf{u}}^h) d\Omega + \int_\Omega (\underline{\boldsymbol{\sigma}}^* - \boldsymbol{\sigma}^h)^T \mathbf{D}^{-1} (\underline{\boldsymbol{\sigma}}^* - \boldsymbol{\sigma}^h) d\Omega\right]^{1/2} \quad (2.15)$$

Notice that the temporal discretization error has been eliminated in $\mathbf{d}^*$. Now, only spatial discretization error exists between $\underline{\mathbf{u}}^*$ and exact solution $\mathbf{u}$.

2.2.2 Postprocessing in Space

The spatial error in the total energy norm now may be written as

$$\left\|\mathbf{e}_s\right\|_E \approx \left\|\mathbf{u}^* - \underline{\mathbf{u}}^*\right\|_E = \left[\int_\Omega (\dot{\mathbf{u}}^* - \dot{\underline{\mathbf{u}}}^*)^T \rho (\dot{\mathbf{u}}^* - \dot{\underline{\mathbf{u}}}^*) d\Omega + \int_\Omega (\boldsymbol{\sigma}^* - \underline{\boldsymbol{\sigma}}^*)^T \mathbf{D}^{-1} (\boldsymbol{\sigma}^* - \underline{\boldsymbol{\sigma}}^*) d\Omega\right]^{1/2} \quad (2.16)$$

where $\dot{\underline{\mathbf{u}}}^*$ and $\underline{\boldsymbol{\sigma}}^*$ are known as the postprocessed solutions in time; $\dot{\mathbf{u}}^*$ and $\boldsymbol{\sigma}^*$ are the recovered solutions of $\dot{\underline{\mathbf{u}}}^*$ and $\underline{\boldsymbol{\sigma}}^*$ by postprocessing in space.

There are many postprocessing techniques for determining higher order stresses in finite element analysis. For example, the simple nodal averaging technique [68] and the global L_2 projection [69] have been used extensively. However, the superconvergent patch recovery (SPR) method introduced by Zienkiewicz and Zhu [7, 8] provides increased accuracy of error estimation with less computational expense [70]. The SPR method exploits the well-known superconvergence properties of the stresses at certain sampling points. The SPR method will be briefly reviewed in Chapter 4. Many others attempted to further improve the performance of SPR method by considering additional fulfillment of local equilibrium and boundary conditions or even element patches instead of node patches [10-12], but the simple procedure originally introduced remains still most

effective [9]. In this study, the original SPR method is implemented on 4-node quadrilateral elements.

As there is no recovered solution $\underline{\dot{\mathbf{u}}}^*$ available, equation (2.16) is simplified as

$$\left\|\mathbf{e}_s\right\|_E \approx \left\|\underline{\mathbf{e}}_s\right\|_E = \left[\int_\Omega (\boldsymbol{\sigma}^* - \underline{\boldsymbol{\sigma}}^*)^T \mathbf{D}^{-1} (\boldsymbol{\sigma}^* - \underline{\boldsymbol{\sigma}}^*) d\Omega\right]^{1/2} \tag{2.17}$$

This simplification is validated by [37] since the error in total energy norm is mainly due to the strain energy of the error at most stages if a sufficiently small time step size is chosen. But in the case where the strain energy is very small or the kinetic energy is dominating, this simplification is invalid.

The relative local temporal error and relative spatial error are defined accordingly as

$$\bar{\eta}_t = \frac{\left\|\underline{\mathbf{e}}_t\right\|_E}{\left\|\mathbf{u}^h\right\|_E} \times 100\% \qquad \text{and} \qquad \bar{\eta}_s = \frac{\left\|\underline{\mathbf{e}}_s\right\|_E}{\left\|\mathbf{u}^h\right\|_E} \times 100\% \tag{2.18a,b}$$

In the denominators of equations (2.18), Wiberg and Li [45] used $(\left\|\mathbf{u}^h\right\|_E)_{\max}$, the maximum total energy norm of the finite element solutions recorded during the computational process. However, their strategy accumulates errors rapidly for systems with damping whose total energy decreases rapidly in time.

Chapter 3: MESH SUPERPOSITION TECHNIQUE

In this chapter, the basic formulation of mesh superposition technique by Fish [15] is reviewed in section 3.1 and the proper DOF suppression is presented in section 3.2 to ensure *compatibility* and *uniqueness* conditions for displacement field. The structured mesh superposition is presented in section 3.3, and the difficulty in calculation of coupling matrices is effectively resolved with the help of the *global-local transformation matrix* C in section 3.4. In section 3.5, the multi-level mesh superposition is presented for the sake of practical applications. In section 3.6, the equation conditioning is investigated for the composite system of equations generated in s-refinement. Finally, in section 3.7, a Diagonal Block Preconditioned Conjugate Gradient equation solver is used to solve the composite system of equations and the numerical example shows the solver is very efficient.

3.1 Hierarchical Formulation

For efficient finite element analysis of large-scale problem, local refinement techniques should be utilized to reduce discretization errors in local domains with the high solution gradients. Various procedures exist for the refinement of finite element solutions [5, 13]. The discretization errors can be reduced by either reducing the sizes of elements (h-refinement), increasing polynomial order of interpolation functions (p-refinement),

relocating the nodes (r-refinement), or by some combination of these techniques (hp-refinement or hr-refinement).

In designing an adaptive strategy, it is critical to consider economy, i.e., to obtain a specified accuracy with a minimal computational work. If the computational work is measured in terms of the accumulative CPU time, *hierarchiality* has a crucial impact on the performance of an adaptive algorithm [25]. Here, hierarchiality means that previously computed information (displacement solutions, system matrices and so on) in solving lower-level idealizations is exploited in solving a higher-level idealization.

In general, h-, r-, hp- and hr-refinement do not possess hierarchiality property due to the optimization in mesh size or the change of nodal locations. Furthermore, the application of these refinement methods in refining local regions makes the use of either special transition elements or multi-point constraints, or the use of the complicated mesh generator inevitable. On contrary, *mesh superposition technique* (hereafter referred to as *s-refinement*) is a mesh refinement technique which attempts to construct a nearly optimal discretization in a purely hierarchical fashion. Mesh superposition technique creates refined subdomains within an existing finite element mesh (hereafter referred to as *global mesh*) by superimposing a sequence of independent, refined meshes (hereafter referred to as *overlay meshes* or *local meshes*) on a portion of the existing finite element to form a composite mesh. The superimposed refined meshes can be constructed by h-, p-

or hp-refinement independently. The hierarchiality property of s-refinement comes from the fact that no changes are made to the original global mesh in mesh superposition process. The global mesh and the overlay meshes need not have compatible discretization, thus the overlay meshes can be used to provide enhanced interpolation precisely where and how they are needed, regardless of the original global mesh topology.

The mechanics of finite element mesh superposition can be best illustrated by considering a simple example such as 2-D, static, displacement-based finite element model of a linear elastic solid shown in Figure 3.1(a). The global mesh is a 4×3 coarse mesh of four-node quadrilateral elements and occupies a problem domain Ω with boundary Γ (i.e. the entire rectangle ABCD). Suppose unacceptable errors have been detected and therefore refinement is needed within the shaded subdomain Ω^L enclosed by boundary Γ^L (i.e. the quadrilateral domain EFGD). Instead of reforming the global mesh on Ω, the level of refinement is increased within subregion Ω^L by constructing an independent overlay mesh (local mesh) corresponding to subregion Ω^L and superimposing it on the global mesh to form a composite mesh. In Figure 3.1(a), a 5×5 overlay mesh of four-node quadrilateral elements occupying Ω^L is constructed and superimposed on the 4×3 coarse global mesh. The resulting composite mesh exhibits increased refinement precisely where it is needed, without the use of troublesome 2-D transition zones.

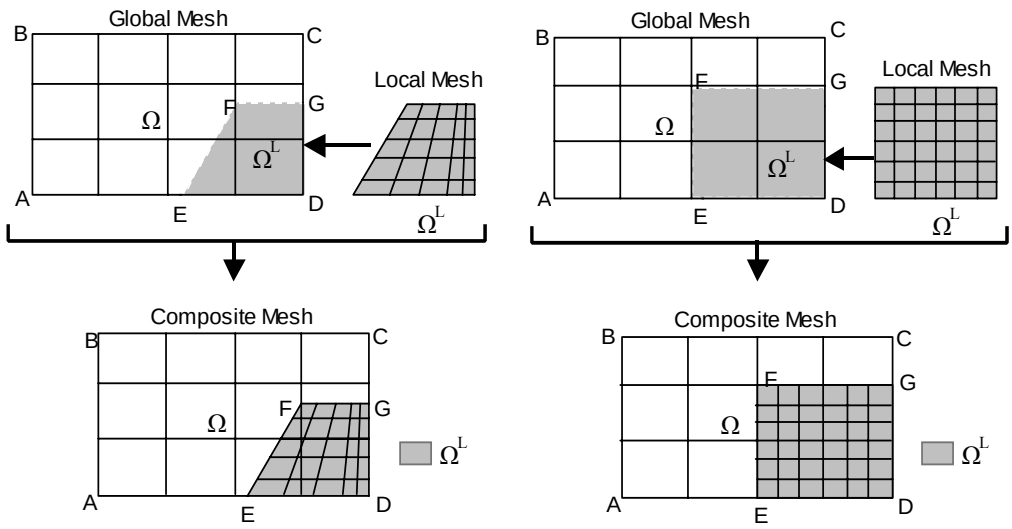


Figure 3.1: A simple example of 2-D finite element mesh superposition

On the composite mesh, the assumed displacement field interpolated on the global mesh is termed as *global displacement* while the one interpolated on the local mesh is termed as *local displacement*. Obviously, the local displacement is zero outside the local mesh. The *total displacement*, or the assumed displacement field for the composite mesh is defined as the sum of the global displacement and the local displacement. For the problem shown in Figure 3.1(a), this definition is expressed as

$$\mathbf{u} = \mathbf{u}^G + \mathbf{u}^L = \mathbf{N}^G\mathbf{d}^G + \mathbf{N}^L\mathbf{d}^L \tag{3.1a}$$

$$\boldsymbol{\varepsilon} = \mathbf{B}^G\mathbf{d}^G + \mathbf{B}^L\mathbf{d}^L \tag{3.1b}$$

where $\mathbf{u}$ is the total displacement vector; $\mathbf{N}$ and $\mathbf{B}$ are the displacement shape function matrix and strain shape function matrix, respectively; $\mathbf{d}$ is the vector of nodal degrees of freedoms, superscripts *G* and *L* denote the quantities associated with the global mesh and

the local mesh, respectively. The displacement vector $\mathbf{u}^L$ serves as a relative displacement; that is, $\mathbf{u}^L$ functions as a refined, incremental enhancement to $\mathbf{u}^G$ within Ω^L. Also note the functions $\mathbf{N}^G$, $\mathbf{B}^G$ and functions $\mathbf{N}^L$, $\mathbf{B}^L$ may be defined on different coordinate systems of global mesh and local mesh.

The s-refinement enriches the solution space locally in exactly the same way as in the conventional h-, p- or hp-refinement, except that the only difference is that the solution space is locally enriched in a hierarchical manner by locally superimposing either finer mesh, higher order elements, or by adopting both simultaneously. Therefore, the s-refinement is able to conveniently provide the same refinement capabilities as h-, p- and hp-refinement. In addition, the mesh superposition technique provides one of its most advantageous features, the possibility of distortion-free local refinement without introducing special transition elements or multi-point constraints.

The development of governing finite element equations in s-refinement proceeds similarly to other displacement-based finite element models. The governing finite element equations of the example problem can be obtained by the minimization of the following total potential energy functional.

$$\Pi(\mathbf{u}) = \frac{1}{2}\int_{\Omega} \boldsymbol{\varepsilon}^T \mathbf{D} \boldsymbol{\varepsilon} d\Omega - \int_{\Omega} \mathbf{u}^T \mathbf{f} d\Omega - \int_{S^\sigma} \mathbf{u}^T \mathbf{T} dS \tag{3.2}$$

where Ω represents the problem domain, S^{σ} is the surface where the external force is applied, $\boldsymbol{\varepsilon}$ is the strain vector, $\mathbf{u}$ is the displacement vector, $\mathbf{D}$ is the elasticity matrix, $\mathbf{f}$ is the body force vector and $\mathbf{T}$ is the surface force vector.

The substitution of equation (3.1a,b) into equation (3.2) and then the minimization of the functional yield the assembled system of equations as

$$\begin{bmatrix} \mathbf{K}^{GG} & \mathbf{K}^{GL} \\ \mathbf{K}^{LG} & \mathbf{K}^{LL} \end{bmatrix} \begin{Bmatrix} \mathbf{d}^{G} \\ \mathbf{d}^{L} \end{Bmatrix} = \begin{Bmatrix} \mathbf{f}^{G} \\ \mathbf{f}^{L} \end{Bmatrix} \tag{3.5}$$

where the matrices and vectors are given as

$$\mathbf{K}^{GG} = \int_{\Omega} (\mathbf{B}^{G})^{T} \mathbf{D} \mathbf{B}^{G} d\Omega \quad \mathbf{K}^{LL} = \int_{\Omega} (\mathbf{B}^{L})^{T} \mathbf{D} \mathbf{B}^{L} d\Omega \quad \mathbf{K}^{GL} = \int_{\Omega} (\mathbf{B}^{G})^{T} \mathbf{D} \mathbf{B}^{L} d\Omega \tag{3.6a}$$

$$\mathbf{f}^{G} = \int_{\Omega} (\mathbf{N}^{G})^{T} \mathbf{f} d\Omega + \int_{S^{\sigma}} (\mathbf{N}^{G})^{T} \mathbf{T} dS \quad \mathbf{f}^{L} = \int_{\Omega} (\mathbf{N}^{L})^{T} \mathbf{f} d\Omega + \int_{S^{\sigma}} (\mathbf{N}^{L})^{T} \mathbf{T} dS \tag{3.6b}$$

$\mathbf{K}^{GG}$ and $\mathbf{K}^{LL}$ are the standard stiffness matrices corresponding to global mesh and local mesh, respectively. $\mathbf{K}^{LG} = (\mathbf{K}^{GL})^{T}$ represents a *coupling matrix* between global and local degrees of freedom. $\mathbf{f}^{G}$ and $\mathbf{f}^{L}$ represent the standard nodal force vectors due to the external forces associated with global mesh and local mesh, respectively.

The hierarchical structure of the assembled stiffness matrix and nodal force vector is immediately observed in equation (3.5). The global stiffness matrix $\mathbf{K}^{GG}$ and nodal force $\mathbf{f}^{G}$ are intactly contained in the refined composite model. The hierarchical structure of the

system matrices, which can be utilized later to significantly expedite adaptive refinement process, is the other most advantageous feature of the mesh superposition technique.

Like in the standard finite element procedure, the assembled matrices and vectors in equation (3.6) are obtained by calculating element matrices and vectors and then assembling them. On an overlay element domain Ω_e, the governing finite element equations are given as

$$\begin{bmatrix} \mathbf{K}_e^{GG} & \mathbf{K}_e^{GL} \\ \mathbf{K}_e^{LG} & \mathbf{K}_e^{LL} \end{bmatrix} \begin{Bmatrix} \mathbf{d}_e^{G} \\ \mathbf{d}_e^{L} \end{Bmatrix} = \begin{Bmatrix} \mathbf{f}_e^{G} \\ \mathbf{f}_e^{L} \end{Bmatrix} \tag{3.7}$$

The matrices and vectors in equation (3.7) have the similar definitions and physical meanings as their counterparts in equation (3.6) except that the integration domain is the overlay element domain Ω_e instead of the whole problem domain Ω.

Once again, note that $\mathbf{B}^G$ and $\mathbf{B}^L$ may be defined on different coordinate systems of global mesh and local mesh, which poses difficulty in calculating $\mathbf{K}_e^{GL}$. This issue will be addressed in section 3.4.

3.2 Proper DOF Suppression

Certain DOF (Degree of Freedom) on overlay mesh should be suppressed to ensure *compatibility* of the total displacement solution and *uniqueness* of the displacement fields on global mesh and local mesh.

The total displacement field should be continuous throughout composite mesh, which is the requirement of compatibility or C^0 continuity. In Figure 3.1, compatibility requirement can be guaranteed conveniently as long as homogeneous essential (zero) boundary conditions are enforced on $\mathbf{u}^L$ along line EFG. Note that continuity of the displacement field is not a question along line EDG that coincides with part of the boundary of global mesh. As a result, homogeneous essential boundary conditions do not have to be enforced on $\mathbf{u}^L$ along line EDG, unless, of course, line EDG happens to be part of a Diriclit boundary where homogeneous essential boundary conditions are enforced for the specific problem. In summary, to ensure C^0 continuity of the total displacement field throughout composite mesh, homogeneous essential boundary conditions need to be enforced on the overlay mesh interpolant $\mathbf{u}^L$ along those parts of overlay mesh boundary Γ^L that do not coincide with the original mesh boundary Γ^G, and also along those parts of overlay mesh boundary Γ^L that coincide with a Diriclit boundary.

The other issue is the singularity of assembled stiffness matrix in equation (3.5), which is caused by the redundant rigid body modes of local meshes. In other words, there may be more than one set of component vector fields ($\mathbf{u}^G$ and $\mathbf{u}^L$) that can be summed to yield a particular total displacement vector field $\mathbf{u}$. To guarantee the uniqueness of the solution, the nodes of the local mesh that coincide with those of global mesh must be enforced homogeneous essential boundary conditions, which has mathematically justified by Kim

et al. **[71]**. This additional constraint may occur to the nodes on the overlay mesh boundary or to the nodes within the overlay mesh. A close examination of Figure 3.1(a) reveals that there are two overlay nodes in the composite mesh whose DOF is needed to be suppressed: one is located at point D while the other is located near the middle of line DG.

3.3 Structured S-version

Although overlay meshes of any size, shape and discretization level can be superimposed anywhere on the global mesh, in this work, the overlay meshes are created and superimposed in such a way that each individual overlay element is superimposed entirely within the domain of a single global element and partial overlapping of a global element is not allowed (Figure 3.1(b)). This structured pattern of mesh overlays was named *structured* s-version (or s_s-version in [15]) in comparison with the *unstructured* mesh superposition (Figure 3.1(a)) where an overlay element is allowed to fully or partially cover portions of two or more global elements. As seen in next section, structured s-version greatly simplifies the computation of the coupling stiffness matrices ($\mathbf{K}^{GL}$ and $\mathbf{K}^{GL}$). Therefore, we shall focus on the structured mesh superposition. However, it should be noted that in restricting further developments to the structured case, the loss in modeling generality is not severe, but there is a significant gain in programming simplicity and efficiency.

In addition, the form of the overlay will be further restricted by creating an individual overlay mesh for each global element that requires refinement. In this case, the overlay mesh is created by uniform subdivision of the global element in its natural coordinate system under the control of two integer *overlay mesh parameters* (d_h and d_v) in the study of 2-D problems. The overlay mesh that is superimposed on any existing element will be a uniform $d_h \times d_v$ mesh that is constructed in the existing element's natural coordinate system. For example, $d_h=3$ and $d_v=3$ are used in Figure 3.1(b). Also note that the d_h and d_v may be independently chosen according to the refinement requirements in each direction. This type of structured overlay mesh generation does not require a conventional mesh generator since it relies on the geometry of the existing global mesh. This feature makes structured mesh superposition very appealing for transient analysis where efficiency and speed are of paramount importance.

In structured s-refinement, the final composite mesh looks very similar to h-refinement based on element subdivision. However, there are several important distinctions between them. The composite mesh in structured s-refinement is composed of two layers of meshes where the original mesh remains unchanged, while traditional h-refinement produces a completely new single-layer mesh. In order for h-refinement to create a refined subregion, each of the existing global elements in the subregion must be replaced by a group of new smaller elements, thus requiring a reformulation of the global mesh. In addition, h-refinement that is based on subdivision of existing elements requires the

implementation of multipoint constraints to ensure compatibility across the boundary of the refined subregion. Thus, structured s-refinement has several operational advantages over h-refinement that is based on element subdivision.

The structured mesh superposition may generate local nodes which coincide with global nodes. The DOFs associated with the labeled nodes on the local mesh in Figure 3.2 are those suppressed DOFs due to the compatibility and uniqueness requirements.

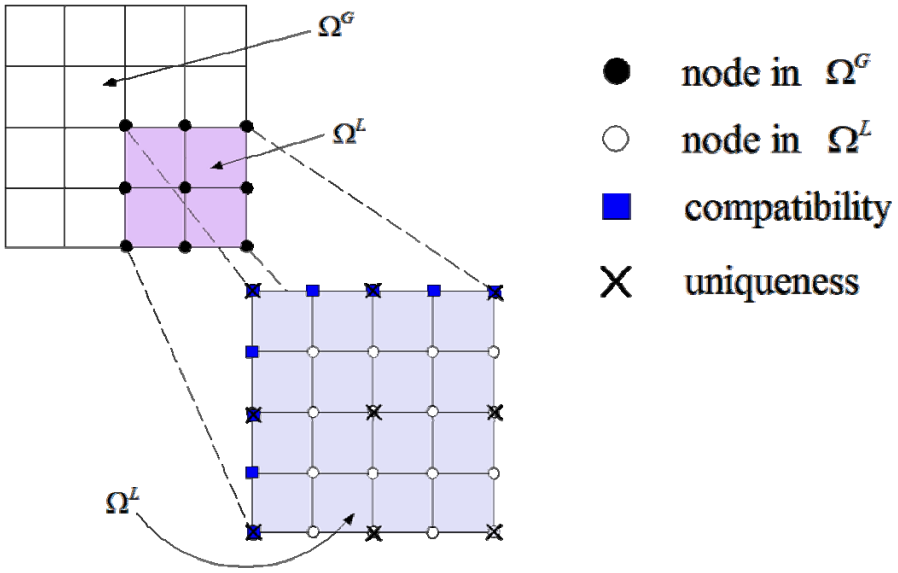


Figure 3.2: Typical DOF suppression in structured mesh superposition

Due to the structured construction of overlay mesh in structured s-refinement, it is simple to perform the inverse mapping from global coordinates (x,y) to the natural coordinates (ξ,η) in the underlying global element of a local node. In Figure 3.3, a global element is superposed by a structured local mesh constructed by parameters d_h and d_v. The local

node under consideration is labeled with ●, which is the intersection point of two straight lines, **Aa** and **Bb**. The length ratio of **3A** over **34** and the length ratio of **1a** over **12** are p/d_h. Similarly, the length ratio of **3B** over **31** and the length ratio of **4b** over **42** are p/d_h. And then, the natural coordinates (ξ,η) of the local node in the underlying global element can be simply calculated as

$$\xi = -1 + \frac{2}{d_h} p\,, \qquad \eta = 1 - \frac{2}{d_v} q \tag{3.8}$$

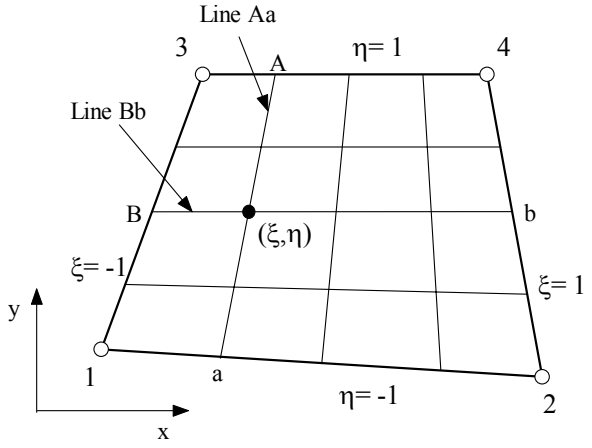


Figure 3.3: The natural coordinates (ξ,η) of a local node in the underlying global element

The computed natural coordinates (ξ,η) will be used in calculating coupling matrix $\mathbf{K}^{GL}$. If local nodes are not determined by the above structured overlay mesh construction, the inverse mapping from coordinates (x,y) to coordinates (ξ,η) necessitates solving a 2×2 system of nonlinear algebraic equations; however, for linear, quadratic and cubic isoparametric elements, the equations are well behaved and thus only 2 or 3 Newton-Raphson iterations are needed to perform the inverse mapping.

3.4 Calculation of Coupling Matrices

We mentioned in section 3.1 that element coupling matrix $\mathbf{K}_e^{GL}$ in equation (3.7) is the result of an integral whose integrand contains products of interpolation functions from the global mesh and the overlay mesh. In computing $\mathbf{K}_e^{GL}$ for *unstructured* mesh superposition, a given overlay element may have to be subdivided into smaller polygons before Gaussian quadrature can be performed [17]. Such a process severely complicates the general computer implementation. But in the *structured* s-refinement in this study, $\mathbf{K}_e^{GL}$ can be computed without explicitly performing any numerical integration due to the fact that the global elements and the overlay elements are of the same type (i.e., same number of nodes, same order of interpolation functions),

Consider the case of a single four-node overlay element Ω_e superimposed on a single four-node global element Ω_E as shown in Figure 3.4, where Ω_e is fully contained in Ω_E. The Cartesian coordinates of the four global nodes and four local nodes are (x_i^E, y_i^E) and (x_i^e, y_i^e) (i=1,2,3,4). $N_{Ei}^G(\xi_E, \eta_E)$ and $N_{ej}^L(\xi_e, \eta_e)$ (i,j=1,2,3,4) denote the interpolation functions defined in global element Ω_E and overlay element Ω_e, respectively. For a point **P** in Ω_e, obviously, **P** is also contained in Ω_E. Therefore, the Cartesian coordinates of point **P** can be interpolated in the global element Ω_E in matrix form or in tensor form as

$$x = \sum_{i=1}^{4} N_{Ei}^{G}(\xi_E, \eta_E) x_i^E = \sum_{i=1}^{4} \left(N_{Ei}^{G}(\xi_E, \eta_E) \right) x_i^E \tag{3.9a}$$

$$x = \mathbf{N}^G \mathbf{x}^E \tag{3.9b}$$

where (ξ_E, η_E) is the natural coordinates of the point **P** in the global element, $\mathbf{N}^G$ is the row vector of the global element's displacement shape functions evaluated at point **P** and $\mathbf{x}^E$ is the column vector of the Cartesian coordinates of the global nodes.

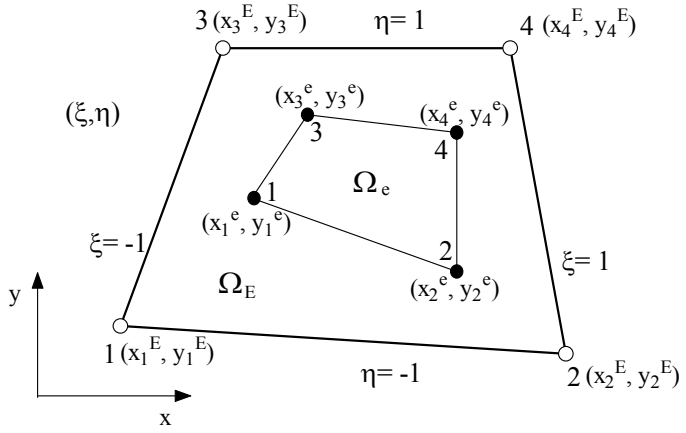


Figure 3.4: An overlay element superimposed within a global element

And similarly, the Cartesian coordinates of point **P** can also be interpolated in the local element Ω_e as

$$x = \sum_{j=1}^{4} N_{ej}^{L}(\xi_e, \eta_e) x_j^e \tag{3.10a}$$

where (ξ_e, η_e) is the natural coordinates of point **P** in the local element and $\mathbf{x}^e$ is the column vector of the Cartesian coordinates of the local nodes.

After interpolating the Cartesian coordinates of the four local nodes in the global element using equation (3.9a), equation (3.10a) can be changed to

$$x = \sum_{j=1}^{4} N_{ej}^{L}(\xi_e, \eta_e)\left(\sum_{i=1}^{4} N_{Ei}^{G}(\xi_{E}^{j}, \eta_E^j) x_i^E \right) = \sum_{i=1}^{4}\left(\sum_{j=1}^{4} N_{ej}^{L}(\xi_e, \eta_e) N_{Ei}^{G}(\xi_E^j, \eta_E^j) \right) x_i^E \quad (3.10b)$$

where the natural coordinates (ξ_E^j, η_E^j) represent the natural coordinates of jth local node in the global element.

After comparing equation (3.9a) and equation (3.10b) and noting the arbitrariness of x and x_i^E, we obtain the relationship in matrix form or in tensor form as

$$N_{Ei}^{G}(\xi_E, \eta_E) = \sum_{j=1}^{4} N_{ej}^{L}(\xi_e, \eta_e) N_{Ei}^{G}(\xi_E^j, \eta_E^j) = \sum_{j=1}^{4} N_{ej}^{L}(\xi_e, \eta_e) C_{ij} \quad (3.11a)$$

$$\mathbf{N}^G = \mathbf{N}^L \mathbf{C}^T \qquad \text{or} \qquad \left(\mathbf{N}^G\right)^T = \mathbf{C}\left(\mathbf{N}^L\right)^T \quad (3.11b)$$

where the components in the matrix **C** can be calculated as

$$C_{ij} = N_{Ei}^{G}(\xi_{E}^{j}, \eta_E^j) \qquad \text{(i,j=1,2,3,4)} \quad (3.12)$$

The matrix **C** is termed as the *global-local transformation matrix*, whose utility is to express the interpolation functions of global element in terms of the interpolation functions of local element as in equation (3.11). Note that the matrix **C** is totally defined by the interpolation function, global element and the local element. A particular matrix **C** works for a particular pair of overlay element and global element. Obviously, the

relationship in equation (3.11) also applies to partial derivatives of the interpolation functions between the global element and the local element as

$$N^G_{Ei,x}(\xi_e,\eta_e)=\sum_{j=1}^{4}C_{ij}N^L_{ej,x}(\xi_e,\eta_e);\quad N^G_{Ei,y}(\xi_e,\eta_e)=\sum_{j=1}^{4}C_{ij}N^L_{ej,y}(\xi_e,\eta_e) \tag{3.13}$$

In the context of finite element method, the strain shape function is obtained by properly differentiating the displacement shape function. Therefore, by using the strain differential operator **L** and then equation (3.11), we have

$$\mathbf{B}^G=\mathbf{L}\mathbf{N}^G=\mathbf{L}(\mathbf{N}^L\mathbf{C}^T)=(\mathbf{L}\mathbf{N}^L)\mathbf{C}^T=\mathbf{B}^L\mathbf{C}^T \tag{3.14}$$

Finally, the element coupling stiffness matrix $\mathbf{K}_e^{GL}$ is obtained as

$$\mathbf{K}_e^{GL}=\int_{\Omega_e}(\mathbf{B}^G)^T\mathbf{D}\mathbf{B}^L d\Omega=\int_{\Omega^e}[\mathbf{B}^L\mathbf{C}^T]^T\mathbf{D}\mathbf{B}^L d\Omega=\mathbf{C}\int_{\Omega^e}[\mathbf{B}^L]^T\mathbf{D}\mathbf{B}^L d\Omega=\mathbf{C}\mathbf{K}_e^{LL} \tag{3.15a}$$

Or written in terms of the components of the matrix as

$$\mathbf{K}_e^{GL}(i,j)=\sum_{k=1}^{4}\mathbf{C}_{ik}\mathbf{K}_e^{LL}(k,j)\quad \text{(i,j=1,2,3,4)} \tag{3.15b}$$

In other words, once the element stiffness matrix $\mathbf{K}_e^{LL}$ is formed for the overlay element Ω_e by standard integration over the overlay element, its corresponding element coupling stiffness matrix $\mathbf{K}_e^{GL}$ can be simply formed by multiplying $\mathbf{K}_e^{LL}$ with the corresponding *global-local transformation matrix* **C** as in equation (3.15). It shall also be noted that the matrix **C** can be similarly utilized to calculate other coupling matrices, such as the coupling mass matrix in dynamic analysis.

Although the 4-node quadrilateral element is taken as the element type for illustration, it is clear that the above derivation does not depend on this specific element type. Therefore, equation (3.15) is also applicable to other types of elements as long as the local element and the global element are isoparametric elements.

3.5 Multi-Level Mesh Superposition

If the structured s-refinement process is restricted to a single layer of overlay meshes, then the discretization of the composite mesh is strongly influenced by the discretization of the global mesh and the overlay mesh parameters d_h and d_v. In order to remove this undesirable dependency, a multi-level structured mesh superposition will be used. In this case, a given composite mesh can be further refined by adding another layer of overlay mesh to the existing composite mesh. This process is illustrated in Figure 3.5 for the case where $d_h=d_v=2$.

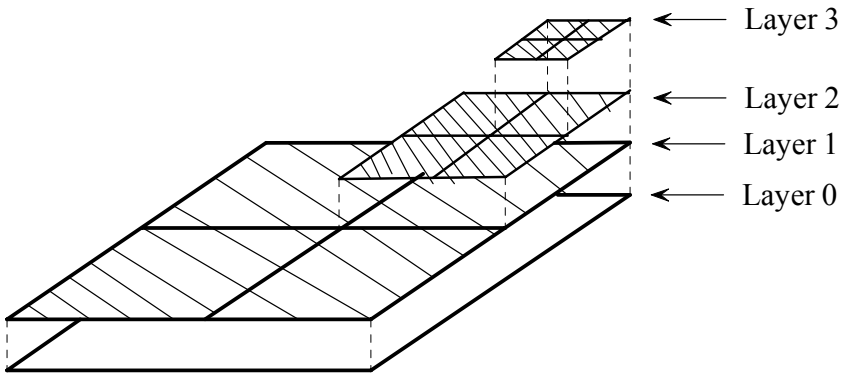


Figure 3.5: Typical structured multi-level mesh superposition

To facilitate further discussion, let us denote the original global mesh as layer 0, while the overlay elements that are superimposed on global elements will constitute layer 1. If further local refinement is needed, then a new set of overlay elements (layer 2) can be created and superimposed on the overlay elements of layer 1. This multi-level refinement process remains *hierarchical* and *structured* in the sense that an element in mesh level *m* must be overlaid entirely within a single element of mesh level *m-1*. The total displacement field is represented by the sum of the displacement field interpolated on the global mesh (layer 0) and the displacement fields interpolated on the local overlay meshes (layer 1,2,3) as in equation (3.16). The extension of mesh superposition method from two-level to multi-level is straightforward and has been successfully applied in linear elastostatics [25, 26].

$$\mathbf{u} = \mathbf{u}^0 + \mathbf{u}^1 + \mathbf{u}^2 + \cdots + \mathbf{u}^n = \mathbf{N}^0\mathbf{d}^0 + \mathbf{N}^1\mathbf{d}^1 + \mathbf{N}^2\mathbf{d}^2 + \cdots + \mathbf{N}^n\mathbf{d}^n \qquad (3.16a)$$

$$\boldsymbol{\varepsilon} = \mathbf{B}^0\mathbf{d}^0 + \mathbf{B}^1\mathbf{d}^1 + \mathbf{B}^2\mathbf{d}^2 + \cdots + \mathbf{B}^n\mathbf{d}^n \qquad (3.16b)$$

where displacement field $\mathbf{u}^i$ is the displacement field interpolated on mesh layer *i* and N^i and d^i are the displacement shape functions matrix and the nodal degrees of freedoms vector on mesh layer *i*, respectively.

Similarly, the governing finite element equations in (3.17) can be obtained by substituting equation (3.16a,b) into the *principles of virtual displacements*.

$$\begin{bmatrix} \mathbf{K}^{00} & \mathbf{K}^{01} & \cdots & \mathbf{K}^{0n} \\ \mathbf{K}^{10} & \mathbf{K}^{11} & \cdots & \mathbf{K}^{1n} \\ \vdots & \vdots & \ddots & \vdots \\ \mathbf{K}^{n0} & \mathbf{K}^{n1} & \cdots & \mathbf{K}^{nn} \end{bmatrix} \begin{Bmatrix} \mathbf{d}^{0} \\ \mathbf{d}^{1} \\ \vdots \\ \mathbf{d}^{n} \end{Bmatrix} = \begin{Bmatrix} \mathbf{f}^{0} \\ \mathbf{f}^{1} \\ \vdots \\ \mathbf{f}^{n} \end{Bmatrix} \tag{3.17}$$

where $\mathbf{K}^{ii}$ denotes the assembled stiffness matrix of *i*th mesh layer. $\mathbf{K}^{ij}$ (i≠j) denotes the assembled coupling stiffness matrix between the DOFs on *i*th mesh layer and the DOFs on *j*th mesh layer.

3.6 Equation Conditioning

It is well known that numerical methods for inverting matrices are very sensitive to the condition number of the matrix, which also gives an indication of the accuracy of the solution from matrix inversion. Zienkiewicz [27] showed that the use of relative variables in a hierarchical finite element formulation produces a better-conditioned system of equations than non-hierarchical forms. In this section, this trend is demonstrated for comparable meshes generated by structured mesh superposition (hierarchical) and traditional h-refinement (non-hierarchical).

The condition numbers for the s-refinement and the h-refinement are calculated and compared for 2-D meshes composed of 4-node quadrilateral elements. Figure 3.6 shows three different global meshes that are used as starting points for the creation of composite meshes using the s-refinement. In each case, the composite mesh is obtained by

superimposing a single layer of overlay mesh on the global mesh. Composite meshes of various discretizations are created by using single overlay meshes characterized by $d_h=d_v=2,4,6,8$, thus producing composite meshes that range in discretization from 2×2 to 32×32 elements. For comparison, the h-refinement is used to produce uniform refined meshes that have the exact same mesh discretization as each of the composite meshes. Table 3.1 compares the condition number of each of the composite meshes (column s) with the condition number of a comparable mesh obtained by the h-refinement (column h), along with the ratio between the condition numbers (column h/s).

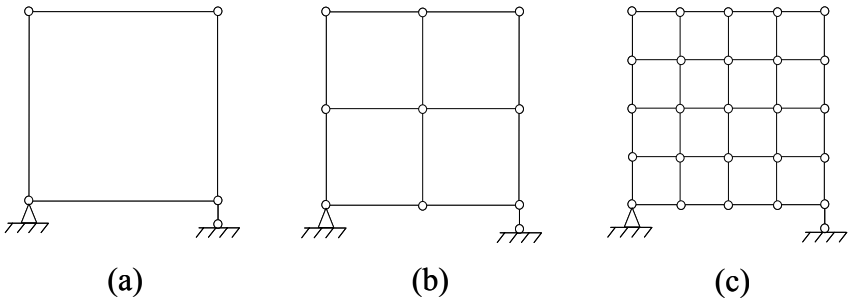


Figure 3.6: A series of global meshes used in the s-refinement

Table 3.1: Comparisons of the condition numbers by s-refinement and h-refinement

	$d_h=d_v=2$			$d_h=d_v=4$			$d_h=d_v=6$			$d_h=d_v=8$		
	s	h	h/s	s	h	h/s	s	h	h/s	s	h	h/s
(a)	50	90	1.8	130	461	3.6	246	1136	4.6	440	2141	4.9
(b)	19	461	2.4	293	2141	7.3	394	5209	13.2	538	9779	18.2
(c)	80	214	2.7	1027	9779	9.5	1157	2375	20.5	1288	4455	34.6

In each case, the s-refinement produces a smaller condition number than h-refinement, indicating a better-conditioned system of equations. Furthermore, the ratio between two condition numbers increases as the mesh density increases. This trend is observed for both increases in the density of the overlay mesh and increases in the density of the global mesh. Thus the s-refinement inherently produces better-conditioned systems of equations than the h-refinement, and the relative superiority of the s-refinement increases with the size of the problem being solved.

The previous comparisons involve composite meshes that are formed by adding a single overlay mesh to the existing global mesh. However, multiple levels of overlay meshes are often required to obtain a composite mesh that is both accurate and efficient. Therefore, let us consider the effect of the number of overlay levels on the condition number of the composite system. Figure 3.7 shows a schematic diagram of three different mesh superposition sequences that are each used to produce a uniform 16×16 two-dimensional composite mesh of 4-node quadrilateral elements. In each case, the global mesh (i.e. level 0) is represented by a single element. The first approach is to set $d_h=d_v=16$ and follow the path shown at the bottom of Figure 3.7 by superimposing a single layer of overlay mesh. The second approach is to set $d_h=d_v=4$ and follow the path shown on the top of Figure 3.7 by superimposing two layers of uniform overlay meshes in a row. The third approach is to set $d_h=d_v=2$ and follow the path shown in the middle of Figure 3.7 by superimposing four layers of overlay meshes successively. The number

attached to each arrow is the condition number of the composite mesh to which the arrow points. Although all three approaches generate composite meshes with the same discretization (16×16), the resulting condition numbers of the three composite meshes are different. The first approach yields 2025; the second approach yields 338 and the third approach yields 289. It is clear that the multi-level mesh superposition has advantage over the two-level mesh superposition in producing a better-conditioned system of equations.

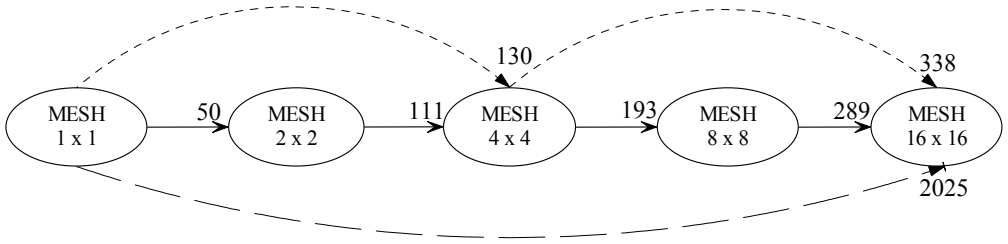


Figure 3.7: Three approaches to achieve a uniform 16x16 composite mesh

3.7 Solution Methodology

Various types of equation solvers have been developed for the finite element analysis, most of which can be classified into two major categories: *direct solvers* and *iterative solvers*. Until recently, direct solvers were often preferred to iterative solvers in real applications because of their robustness and predictable behavior. However, many efficient iterative solvers were discovered and the increased need for solving very large

linear systems triggered a noticeable and rapid shift toward iterative technique in many areas of scientific computing [72]. Also, iterative solvers are gaining ground over direct solvers because they are easier to implement efficiently on high-performance computers.

Although the lack of robustness is the widely recognized weakness of iterative solvers, both robustness and efficiency of iterative solvers can be improved by *preconditioning*. Roughly speaking, a preconditioning matrix, or a *preconditioner* **M**, approximates the original system matrix **K** in any explicit or implicit form to make the original linear system **Kx=f** easier to solve. From a practical point of view, the only requirement for a preconditioner **M** is that it is inexpensive to solve the linear system **Mx=f** because its solution is required at each iteration of the preconditioned algorithms.

$$\begin{bmatrix} \mathbf{K}^{GG} & \mathbf{K}^{GL} \\ \mathbf{K}^{LG} & \mathbf{K}^{LL} \end{bmatrix} =$$

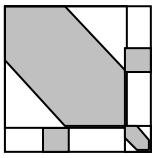


Figure 3.8: Typical profile of the assembled stiffness matrix

Figure 3.8 shows a typical profile of the composite stiffness matrix. While matrices $\mathbf{K}^{GG}$ and $\mathbf{K}^{LL}$ possess banded structure, the composite stiffness matrix does not have a banded structure due to the presence of the coupling matrices $\mathbf{K}^{GL}$ and $\mathbf{K}^{LG}$. Consequently, the composite matrix experiences a considerable increase in the bandwidth, thus resulting in that the commonly used direct, banded solvers become way too expensive in terms of CPU time and memory requirement. In this case, iterative solvers are more appropriate

for three reasons. First, iterative solvers are not directly affected by the increase in the bandwidth of the system matrix. Second, the hierarchical nature of the composite system naturally provides excellent starting vectors for the iterative solution process, namely, the solution vector from a previous lower level solution. Last, the system of equations produced by structured s-refinement is well-conditioned which implies quick convergence and more accurate solution for iterative solvers.

A block iterative method was adopted by Robbins [19], where a 'bandwidth minimizer' was used to renumber the DOFs in the composite mesh, thus obtaining a new rearranged system matrix whose bandwidth is larger than the bandwidth of $\mathbf{K}^{GG}$ or $\mathbf{K}^{LL}$, but much smaller than the bandwidth of composite matrix. It was found out that the block iterative method is even better than the direct, banded solver in solving the rearranged system, in term of total number of arithmetic operations.

Among the most efficient iterative methods, the *Preconditioned Conjugate Gradient (PCG) methods*, whose performance strongly depends on the preconditioner utilized, and the *Multigrid (MG) methods* which require a hierarchy of discretizations, i.e., several discretizations of increasing density of the same problem domain, are the most noteworthy. Fish [24] showed that PCG solver is faster than MG solver, although, for large-scale problems, both methods are vastly superior to direct elimination. Joulard and Coutinho [73] carried out a comparison between PCG and MG solvers in a series of plane

elasticity problems. They found that the hierarchical PCG solver is faster than the MG solver in all problems tested. In this study, equation (3.5) is solved using the PCG method with a diagonal block preconditioner **M** which discards all the coupling matrices $\mathbf{K}^{GL}$ and $\mathbf{K}^{LG}$ in the composite matrix.

$$\mathbf{M} = \begin{bmatrix} \mathbf{K}^{GG} & \mathbf{0} \\ \mathbf{0} & \mathbf{K}^{LL} \end{bmatrix} \tag{3.18}$$

Due to the diagonal block structure of matrix **M**, its inverse matrix is simply given as

$$\mathbf{M}^{-1} = \begin{bmatrix} (\mathbf{K}^{GG})^{-1} & \mathbf{0} \\ \mathbf{0} & (\mathbf{K}^{LL})^{-1} \end{bmatrix} \tag{3.19}$$

where $(\mathbf{K}^{GG})^{-1}$ and $(\mathbf{K}^{LL})^{-1}$ may be calculated efficiently and further stored by direct, banded solvers due to the standard banded structures exhibited by $\mathbf{K}^{GG}$ and $\mathbf{K}^{LL}$.

The above PCG method using a diagonal block preconditioner **M** for the solution of the linear system generated from mesh superposition is called *Diagonal Block PCG* method in this study. Moreover, the diagonal block structure in the composite matrix guarantees that the diagonal block PCG method also applicable to the linear system generated by multi-level mesh superposition.

To show the good performance of the diagonal block PCG method, it is compared with the block iterative method adopted in [19]. The test problem is a stress wave propagation problem in an unconstrained elastic bar. The overlay mesh is repeatedly updated to

provide a composite mesh with high refinement in the immediate vicinity of the traveling stress wave. The two solvers are used to solve exactly the same linear systems of equations at four randomly chosen discrete time points t=0.0090, 0.0220, 0.0345 and 0.0460 sec. The same convergence tolerance ε is used for the two solvers. Table 3.2 records the number of iterations, execution time and their corresponding ratio. The row *ratio* indicates the ratios between the results of the block iterative solver and the ones of the diagonal block PCG solver. The system function *etime* in scientific computing language Fortran90 is to record the elapsed execution time for the two solvers to solve the linear systems of equations. A qualitative analyse of the arithmetic operations involved in the iteration stages of the two solvers reveals that after $(\mathbf{K}^{GG})^{-1}$ and $(\mathbf{K}^{LL})^{-1}$ have been calculated by a direct, banded solver in the first iteration which is necessary for both the solvers in comparison, the arithmetic operations involved in the subsequent iterations are comparable for both the solvers in comparison. This basically indicates that the execution time for each single iteration is comparable for the two solvers. Therefore, the number of iterations serves as an effective indicator of the execution time. In Table 3.2, it is observed that the diagonal block PCG method converges in much less iterations than the block iterative method, and confirmatively, it is observed that less execution time is needed by the diagonal block PCG method. It is also noticed that the ratios of the execution time are integers, which may be caused by the rounding off of the Fortran90 function *etime* in recording the execution time.

Table 3.2: Comparison of performances between the diagonal block PCG method and the block iterative method

	t=0.0090 (s)		t=0.0220 (s)		t=0.0345 (s)		t=0.0460 (s)	
	iter.	time(10^{-2}s)	iter.	time(10^{-2}s)	iter.	time(10^{-2}s)	iter.	time(10^{-2}s)
PCG	22	1.0014415	18	1.0014415	18	1.0014296	20	1.0014534
Iterative	163	8.0115199	132	7.0100784	149	8.0115199	149	8.0115318
Ratio	7.41	8.00	7.33	7.00	8.2	8.00	7.45	8.00

Chapter 4: SUPERCONVERGENT PATCH RECOVERY TECHNIQUE AND THE ZIENKIEWICZ-ZHU (Z^2) ERROR ESTIMATOR

In this chapter, the superconvergent patch recovery (SPR) technique of Zienkiewicz & Zhu [7] is first reviewed in section 4.1, and then is applied to the Zienkiewicz-Zhu (Z^2) error estimator [8] in section 4.2. In section 4.3, the rank deficiency problem in the SPR procedure with 4-node quadrilateral elements is discussed and a *rotated SPR* procedure is developed to simply and efficiently solve the rank deficiency problem by utilizing a local rotated coordinate system. In section 4.4, a new *Layer-by-Layer SPR* scheme is developed to utilize the standard SPR technique in the composite meshes that are generated during the structured mesh superposition.

4.1 Superconvergent Patch Recovery Technique

The Superconvergent Patch Recovery (SPR) technique of Zienkiewicz & Zhu [7, 8] was a very significant breakthrough for quality-assessed FE solutions. The locally improved stress fields that are computed via SPR during postprocessing of the finite element solution are typically of such good quality that they can be used to develop high quality error estimators [70]. In addition, the computational cost of the SPR technique is quite low. These two features (accuracy and low cost), coupled with high convergence rates,

have resulted in widespread use and acceptance of SPR in finite element simulations in many different application areas.

It is well known that the displacement-based finite element method typically results in a stress field $\boldsymbol{\sigma}_h$ that exhibits reduced accuracy and discontinuity at inter-element boundaries. The objective of the SPR technique is to find a smoothed, continuous stress field $\boldsymbol{\sigma}^*$, which is more accurate than $\boldsymbol{\sigma}_h$ and is expressed in terms of the displacement shape functions $\mathbf{N}$ and the recovered nodal stresses $\overline{\boldsymbol{\sigma}}^*$ as

$$\boldsymbol{\sigma}^* = \mathbf{N}\overline{\boldsymbol{\sigma}}^* \tag{4.1}$$

For most types of finite elements, there exist certain sampling points within the element where the raw computed finite element stresses $\boldsymbol{\sigma}_h$ havea greater accuracy and a higher order of convergence than elsewhere [74]. Accordingly, these sampling points are referred to as *superconvergent sampling points*. For example, a 2-D 4-node quadrilateral element contains a single superconvergent sampling point which coincides with the element's centroid. The SPR procedure is based on a least square fit of a higher-order stress distribution governed by the finite element stresses at the superconvergent sampling points.

The first step in the SPR procedure involves selecting a particular vertex node (referred to as *assembly node*) and then identifying the collection of elements (referred to as *element patch*) that are connected to the vertex node. A typical assembly node and the

associated element patch are shown in Figure 4.1 for a 2-D mesh composed of 4-node quadrilateral elements. Obviously, the element patch contains four elements and subsequently contains four sampling points.

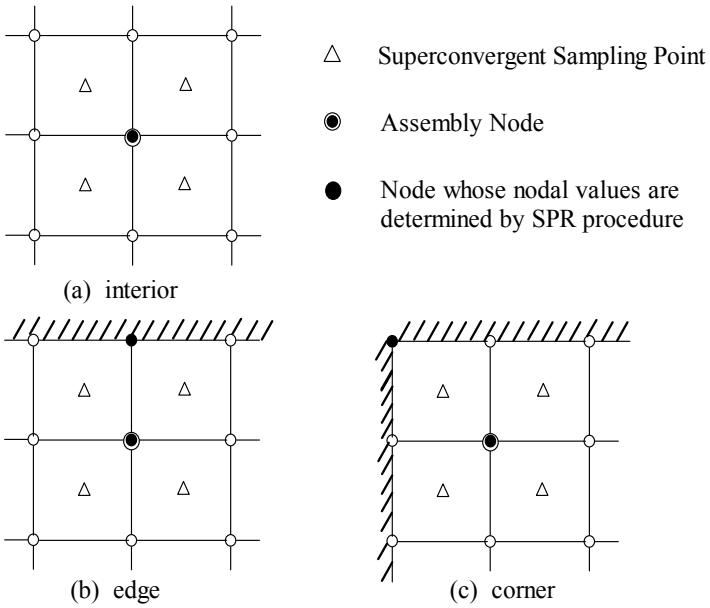


Figure 4.1: Superconvergent patches for a 4-node quadrilateral element mesh

Next, the finite element stresses $\boldsymbol{\sigma}_h$ are computed at the superconvergent sampling points within each element of the element patch and are used to define a continuous local stress field $\boldsymbol{\sigma}_p^*$ within the element patch. Moreover, the local stress field $\boldsymbol{\sigma}_p^*$ is expressed as a complete polynomial of the same order as used in the interpolation of the displacement.

$$\boldsymbol{\sigma}_p^* = \mathbf{Pa} \tag{4.2}$$

where **P** contains the appropriate polynomial terms and **a** is a set of unknown variables. For example, $\mathbf{P} = [1, x, y, xy]$ and $\mathbf{a} = [a_1, a_2, a_3, a_4]^{\mathbf{T}}$ can be chosen for the 4-node

quadrilateral element. In the least squares fit of $\boldsymbol{\sigma}_p^*$ to $\boldsymbol{\sigma}_h$ at the superconvergent sampling points, the unknown variables $\mathbf{a} = [a_1, a_2, a_3, a_4]^{\mathbf{T}}$ are determined by minimizing the following discrete least squares functional:

$$\Pi(\mathbf{a}) = \sum_{i=1}^{n} (\boldsymbol{\sigma}^{\mathrm{h}}(x_i, y_i) - \boldsymbol{\sigma}_p^*(x_i, y_i))^2 = \sum_{i=1}^{n} (\boldsymbol{\sigma}^{\mathrm{h}}(x_i, y_i) - \mathbf{P}(x_i, y_i)\mathbf{a})^2 \tag{4.3}$$

where $n=mk$ is the number of superconvergent sampling points in the patch, m denotes the number of elements in the patch and k denotes the typical number of superconvergent sampling points in each element. For a 4-element patch of 4-node quadrilateral elements in Figure 4.1, m=4, k=1 and n=4. And (x_i,y_i) are the coordinates of the ith superconvergent sampling points in the patch. The minimization of $\Pi(\mathbf{a})$ with respect to variable $\mathbf{a}$ yields the following linear algebraic system of equations:

$$\sum_{i=1}^{mk} \mathbf{P}^{\mathrm{T}}(x_i, y_i)\mathbf{P}(x_i, y_i)\mathbf{a} = \sum_{i=1}^{mk} \mathbf{P}^{\mathrm{T}}(x_i, y_i)\boldsymbol{\sigma}_h(x_i, y_i) \tag{4.4}$$

The linear algebraic system of equations in (4.4) can be solved in matrix form

$$\mathbf{a} = \mathbf{A}^{-1}\mathbf{b} \tag{4.5a}$$

$$\mathbf{A} = \sum_{i=1}^{mk} \mathbf{P}^{\mathrm{T}}(x_i, y_i)\mathbf{P}(x_i, y_i), \qquad \mathbf{b} = \sum_{i=1}^{mk} \mathbf{P}^{\mathrm{T}}(x_i, y_i)\boldsymbol{\sigma}_h(x_i, y_i) \tag{4.5b,c}$$

The system matrix $\mathbf{A}$ in equation (4.5b) is purely geometric and independent of stresses; and it needs to be computed only once for all stress components. The number of equations to solve in equation (4.5a) is determined by the number of terms in $\mathbf{P}$, which is

modest in general. The SPR procedure is totally localized on one element patch, which indicates no expensive global system needs to be built or solved. Therefore, the procedure is quite computationally inexpensive.

Once the variable **a** is determined from (4.5a), the recovered nodal stresses $\overline{\boldsymbol{\sigma}}^*$ are simply calculated by inserting appropriate coordinates into the expression for $\boldsymbol{\sigma}_p^*$ as

$$\overline{\boldsymbol{\sigma}}^*(x_i, y_i) = \boldsymbol{\sigma}_p^*(x_i, y_i) = \mathbf{P}(x_i, y_i)\mathbf{a} \tag{4.6}$$

where (x_i,y_i) are the coordinates of the nodes labeled with ● in the element patch as shown in Figure 4.1.

Although discussed here for the specific case of 4-node quadrilateral elements, the SPR procedure can be effectively applied to a wide range of elements. The numerical experiments [7] demonstrate that the recovered nodal stresses with linear and cubic elements are superconvergent (one order higher accuracy) and the recovered nodal stresses with quadratic elements are ultraconvergent (two orders higher accuracy).

A more difficult situation arises at the domain boundary where an element patch may include only one or two elements (Figure 4.1(b,c)). For the *one element situation* (corner node), the number of elements in the patch is insufficient to determine the variable vector **a** and therefore the corner node values are determined from an interior patch (Figure 4.1(c)). For *two-element situation* (edge node), the patch is assembled by using two more

interior elements adjacent to the two boundary elements (Figure 4.1(b)). Zienkiewicz and Zhu [7] claimed that the recovered values of all boundary nodes can be determined by interior patches instead of boundary patches and the results are equally accurate.

4.2 Zienkiewicz-Zhu (Z^2) Error Estimator

The postprocessed/recovered stresses by the SPR technique are used to compute *a posteriori* Zienkiewicz-Zhu (Z^2) error estimator [7, 8]. In the example problem of elasticity, the error of the finite element displacement $\mathbf{u}^h$ with respect to the exact solution $\mathbf{u}$ is defined as

$$\mathbf{e} = \mathbf{u} - \mathbf{u}^h \tag{4.7}$$

And the stress error is defined as

$$\mathbf{e}_\sigma = \boldsymbol{\sigma} - \boldsymbol{\sigma}^h \tag{4.8}$$

The errors in equation (4.7, 4.8) are not practically accessible due to the general unavailability of the exact solutions. However, with the more accurate stresses that are recovered by the SPR procedure, the errors are evaluated simply by replacing the exact solutions in equation (4.8) by the more accurate recovered stresses. The Z^2 error estimator uses the difference between the recovered stress $\boldsymbol{\sigma}^*$ and the raw finite element stress $\boldsymbol{\sigma}^h$ as a measure of error.

$$\mathbf{e}_\sigma^* = \boldsymbol{\sigma}^* - \boldsymbol{\sigma}^h \tag{4.9}$$

The error can be evaluated in any appropriate norm. The most frequently used norm is the energy norm, which is adopted in present study. In this case, the energy norm of $\mathbf{e}_\sigma^*$ can be expressed as

$$\left\|\mathbf{e}\right\|_{\mathrm{E}} = \left\|\mathbf{u}^* - \mathbf{u}^{\mathrm{h}}\right\|_{\mathrm{E}} = \left[\int_\Omega (\boldsymbol{\sigma}^* - \boldsymbol{\sigma}^{\mathrm{h}})^{\mathrm{T}} \mathbf{D}^{-1} (\boldsymbol{\sigma}^* - \boldsymbol{\sigma}^{\mathrm{h}}) d\Omega\right]^{1/2} = \left[\sum \int_{\Omega^i} (\boldsymbol{\sigma}^* - \boldsymbol{\sigma}^{\mathrm{h}})^{\mathrm{T}} \mathbf{D}^{-1} (\boldsymbol{\sigma}^* - \boldsymbol{\sigma}^{\mathrm{h}}) d\Omega\right]^{1/2} \quad (4.10)$$

where Ω is the problem domain, Ω^i is the element domain, $\mathbf{D}$ is the elasticity matrix.

The superconvergent property of the recovered stresses in the SPR procedure guarantees that the Z^2 error estimator is asymptotically exact in the energy norm [36]. It is noteworthy that this desirable characteristic may not be true for the residual type error estimators.

In summary, with the superconvergent accuracy achieved by the SPR procedure and the economy in its implementation, the Z^2 error estimator has several advantages over other types of estimators: simplicity of implementation, cost effectiveness and asymptotical exactness.

4.3 Rank Deficiency in SPR procedure with 4-Node Quadrilateral Elements

The success of the SPR technique has been demonstrated for a wide range of element types in the adaptive analysis of static or dynamic problems in one, two and three

dimensions [7, 8, 14, 45, 65]. Further enhancements of the SPR technique were attempted by taking into account the additional constraints such as the residual of equilibrium and the imbalance in boundary conditions [10, 11, 75], but the original simple SPR procedure remains still most effective [9]. However, Labbe and Garon [76] pointed out that the SPR method might have two problems concerning the local system of equations generated for the recovery patch. These are *rank deficiency* and *bad conditioning* of the system of equations for high-order elements. Zienkiewicz et al. [77] further introduced the use of locally normalized coordinates, which improved the conditioning of the local system and enhanced the SPR procedure, making it more amenable to high-order elements. However, the problem of rank deficiency or singularity of the local system was encountered under certain purely geometric conditions in the present implementation of the SPR procedure with 4-node 2-D quadrilateral elements. A simple strategy of using a local rotated coordinate system is proposed to effectively solve the rank deficiency problem, thus making the SPR procedure robust for use in general automated adaptive procedures.

4.3.1 Rank Deficiency in Local System with Quadrilateral Elements

For the linear algebraic system in equations (4.5) to be uniquely solvable, the rank (or the maximum number of independent rows/columns) of the system matrix **A** has to be at least equal to the number of terms in the polynomial **P** expansion. This implies

$$mk \geq t \tag{4.11}$$

where m is the number of elements in a patch, k is the number of sampling points per element and t is the number of terms in the polynomial **P**. For the finite element mesh of 4-node quadrilateral elements, there are four terms in **P** (t=4), four elements in one patch (m=4) and one sampling point per element (k=1). The contribution of a single sampling point (x_i,y_i) (i=1,2,3,4) to the local system matrix **A** is the component matrix $\mathbf{A}_i$ given as

$$\mathbf{A}_i = \begin{bmatrix} 1 & x_i & y_i & x_i y_i \\ x_i & x_i^2 & x_i y_i & x_i^2 y_i \\ y_i & x_i y_i & y_i^2 & x_i y_i^2 \\ x_i y_i & x_i^2 y_i & x_i y_i^2 & x_i^2 y_i^2 \end{bmatrix} \tag{4.12}$$

Obviously, the rank of matrix $\mathbf{A}_i$ is 1. In other words, $\mathbf{A}_i$ can provide only one independent relationship. In order for the patch of quadrilateral elements to satisfy (4.11), each of the four sampling points should provide exactly one independent relationship; otherwise, **A** will be rank deficient or singular, precluding a unique solution for the linear algebraic system of equations (4.5).

The local system matrix **A** in 4.5(b) is totally determined by the geometrical locations of the sampling points once a polynomial expansion **P** is chosen. Therefore, it is possible that a specific spatial distribution of the four sampling points could prevent the four component matrices $\mathbf{A}_i$ (i=1,2,3,4) from supplying four independent relationships, thus

resulting in a rank deficient matrix **A**. Figure 4.2 shows a scenario in which this rank deficiency occurs.

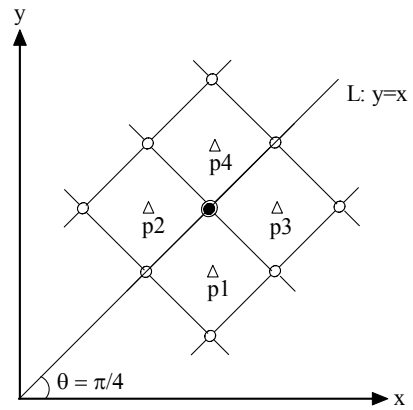


Figure 4.2: A typical configuration of a superconvergent patch of 4-node quadrilateral elements with rank deficiency in its local system

In Figure 4.2, the four superconvergent sampling points p_1, p_2, p_3 and p_4 in the SPR patch are labeled with Δ. This spatial distribution of the sampling points, characterized by sampling points which exhibit symmetry about a 45° line, results in a rank deficient local system matrix **A**. The rigorous mathematical proof of this assertion is tedious and lengthy, but can be numerically verified by calculating the rank number or the condition number of the local system matrix **A.** In this case, the matrix **A** has a rank number of 3 (one less than the full rank of 4) and its condition number approaches infinity.

The rank deficiency case in Figure 4.2 can be generalized. Consider the family of lines in Figure 4.3(a) which has the form x+y=k and the family of lines in Figure 4.3(b) which has form x-y=k. If the set of four sampling points in one patch are symmetrically

distributed about any of the lines x+y=k or x−y=k, then the local system matrix **A** will be rank deficient. This generalized conclusion is verified numerically for the patch shown in Figure 4.4 under a counter-clockwise rotation of coordinate system from 0° to 360°.

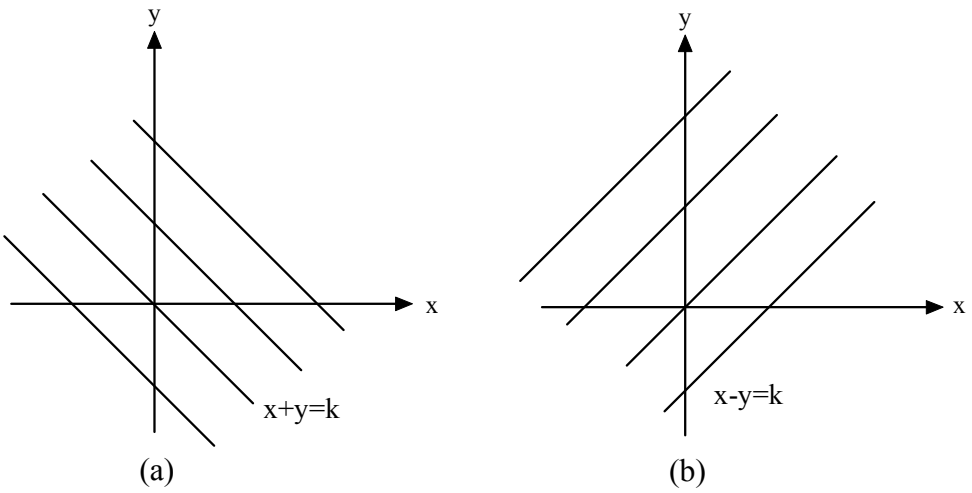


Figure 4.3: (a) The family of lines: x+y=k; (b) the family of lines: x-y=k

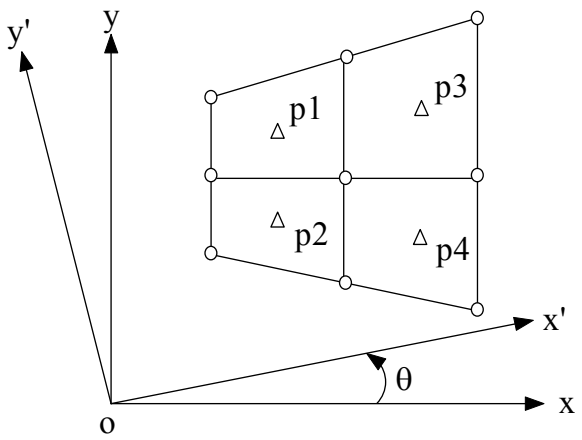


Figure 4.4: An SPR patch of 4-node quadrilateral elements under the rotation of coordinate system

After the rotation of angle θ, the original coordinate system (x,y) becomes the new coordinate system (x',y') and the local matrix **A** is calculated in the new coordinate system. When θ equals 45°, 135°, 225° or 315°, the four sampling points exhibit a symmetric distribution about a line of the form x'+y'=k or x'–y'=k in the new coordinate system, resulting in a singular matrix **A**. Figure 4.5 shows the condition number of the local system matrix **A** as a function of the coordinate system rotation angle θ. We observe that the condition number of the matrix **A** increases sharply implying singularity as θ approaches 45°, 135°, 225° or 315°.

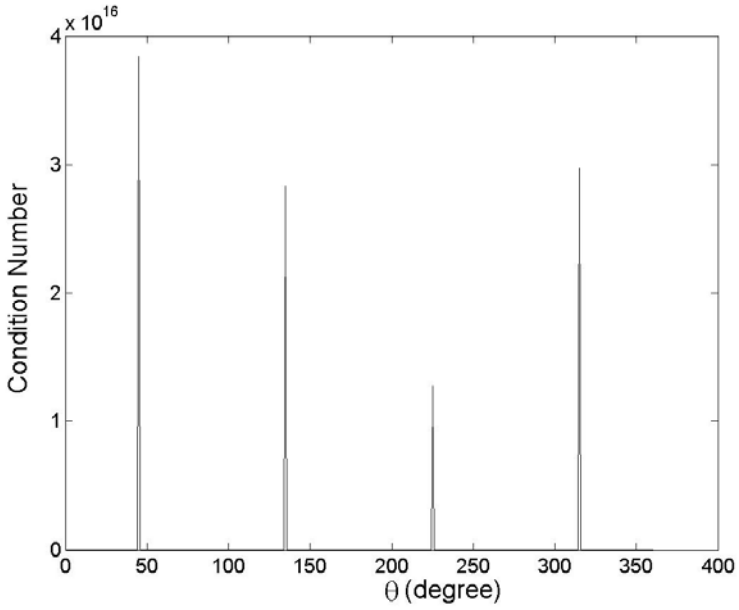


Figure 4.5: Change of condition number with respect to θ when rotating the coordinate system

Figure 4.6 shows a close-up of the condition number of the matrix **A** for coordinate system rotation angles in the range -45°≤θ≤45°. The condition number approaches

infinity when θ=-45° or 45°; and achieves a minimum when θ=0°, which is the optimal angle for accurately solving the linear algebraic equation system in (4.5).

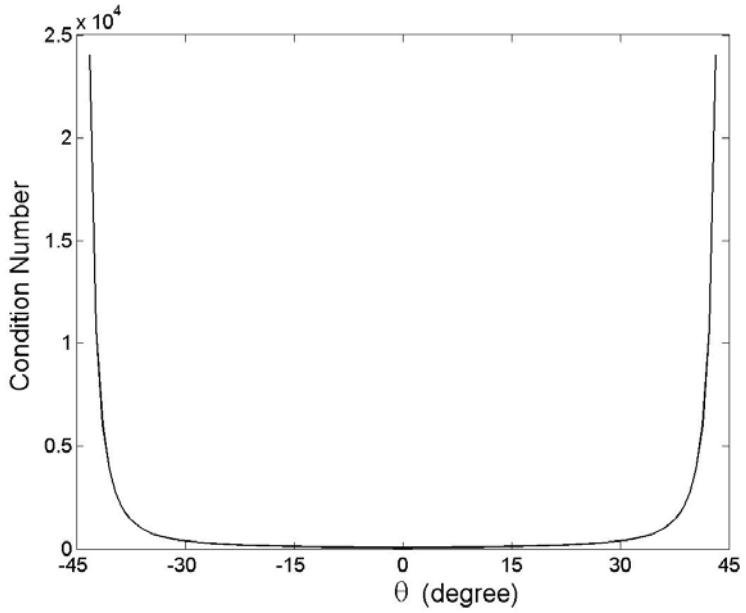


Figure 4.6: Change of condition number with respect to θ (-45°≤θ≤45°)

4.3.2 The Rotated SPR Procedure

Due to the directional dependence of the condition number of the local system matrix **A** in equation (4.5b), a simple rotation of coordinate system can be used to correct any rank deficiency that is caused by an unfortunate symmetric distribution of sampling points. Inspired by the fact shown in Figure 4.6 that a coordinate system rotation of 45° will restore full rank to the rank deficient matrix **A** and result in a minimal condition number, a *rotated SPR* procedure is developed in this study to solve the rank deficiency problem.

After detecting the rank deficiency in matrix **A**, the *rotated SPR* procedure first requires a coordinate system rotation of 45°. Consequently, the transformations of the sampling point coordinates $(x_i,y_i)\rightarrow(x'_i,y'_i)$ and the sampled stress components $(\sigma_{xx}, \sigma_{yy}, \sigma_{xy})\rightarrow(\sigma_{x'x'}, \sigma_{y'y'}, \sigma_{x'y'})$ are performed where the primed quantities indicate direction in the rotated coordinate system. These transformations are performed via equations (4.13) and (4.14) using $\theta=45°$. The rotation of the coordinate system restores the full rank to the rank deficient matrix **A** and thus the standard SPR procedure can be performed in the rotated coordinate system. The linear algebraic system of equations in equation (4.5) is assembled and solved with full rank, yielding coefficients $\mathbf{a} = [a_1, a_2, a_3, a_4]^{\mathbf{T}}$ that can then be used to compute recovered nodal stresses at the desired vertex node via equation (4.2). Since the nodal stresses are expressed in the rotated coordinate system (x',y'), equation (4.14) is then used to transform the nodal stresses back to the global coordinate system (x,y) via a reverse rotation of $\theta= -45°$. The two transformations involved in the *rotated SPR* procedure do not add significantly to the computational burden because they are extremely simple and only applied locally in the patch of deficient rank.

$$x' = x\cos\theta + y\sin\theta \tag{4.13a}$$

$$y' = -x\sin\theta + y\cos\theta \tag{4.13b}$$

$$\sigma_{x'} = \frac{\sigma_x + \sigma_y}{2} + \frac{\sigma_x - \sigma_y}{2}\cos 2\theta + \tau_{xy}\sin 2\theta \tag{4.14a}$$

$$\sigma_{y'} = \frac{\sigma_x + \sigma_y}{2} - \frac{\sigma_x - \sigma_y}{2}\cos 2\theta - \tau_{xy}\sin 2\theta \tag{4.14b}$$

$$\tau_{x'y'} = -\frac{\sigma_x - \sigma_y}{2}\sin 2\theta + \tau_{xy}\cos 2\theta \qquad (4.14c)$$

4.3.3 Numerical Example

The *rotated SPR* procedure is demonstrated in this section on the example of an infinite plate with a central circular hole under uniaxial tensile loads. The geometry, material properties, constraints and external loading are given in Figure 4.7(a). A state of plane strain is assumed. This infinite plate problem has an analytical exact stress distribution given by equations (4.13a,b,c), which is used to define traction boundary conditions on edges BC and CD of the truncated computational domain in Figure 4.7(b). Symmetric displacement boundary conditions are specified on edges AB and ED. The 4-node quadrilateral elements are used. The example problem was also analyzed by Zienkiewicz [7], where 9-node quadrilateral elements were used.

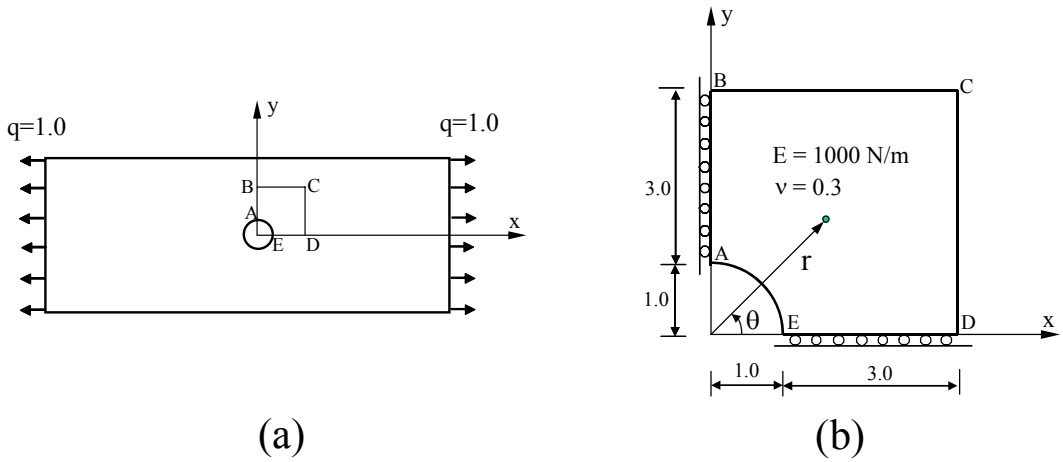


Figure 4.7: Infinite plate with a central circular hole under uniaxial tensile loads: (a) the problem definition; (b) the truncated computational domain

$$\sigma_x = 1 - \frac{r_0^2}{r^2}\left(\frac{3}{2}\cos 2\theta + \cos 4\theta\right) + \frac{3}{2}\frac{r_0^4}{r^4}\cos 4\theta \tag{4.15a}$$

$$\sigma_y = -\frac{r_0^2}{r^2}\left(\frac{1}{2}\cos 2\theta - \cos 4\theta\right) - \frac{3}{2}\frac{r_0^4}{r^4}\cos 4\theta \tag{4.15b}$$

$$\tau_{xy} = -\frac{r_0^2}{r^2}\left(\frac{1}{2}\sin 2\theta + \sin 4\theta\right) + \frac{3}{2}\frac{r_0^4}{r^4}\sin 4\theta \tag{4.15c}$$

where (r,θ) are the polar coordinates and $r_0 = 1$ is the radius of the hole.

Figure 4.8 shows four different meshes of 4-node quadrilateral elements that are used to solve the problem. Also shown in Figure 4.8 is a vertex point **P** with global Cartesian coordinates (x,y) = (1.0606602, 1.0606602), or polar coordinates $(r,\theta) = (1.5, 45^\circ)$, where the exact stress solution has the values of σ_{xx} =1.1481481, σ_{yy} = –0.1481481 and σ_{xy} = –0.22222229. In all four meshes, the vertex point **P** is surrounded by a 4-element patch whose set of four superconvergent sampling points happen to be symmetrically distributed about the line x=y, thus resulting in singularity in local system matrix **A** during the application of the standard SPR procedure. Therefore, the *rotated SPR* procedure is used in each mesh to overcome the rank deficiency problem and thus obtain the recovered stress components at vertex point **P**.

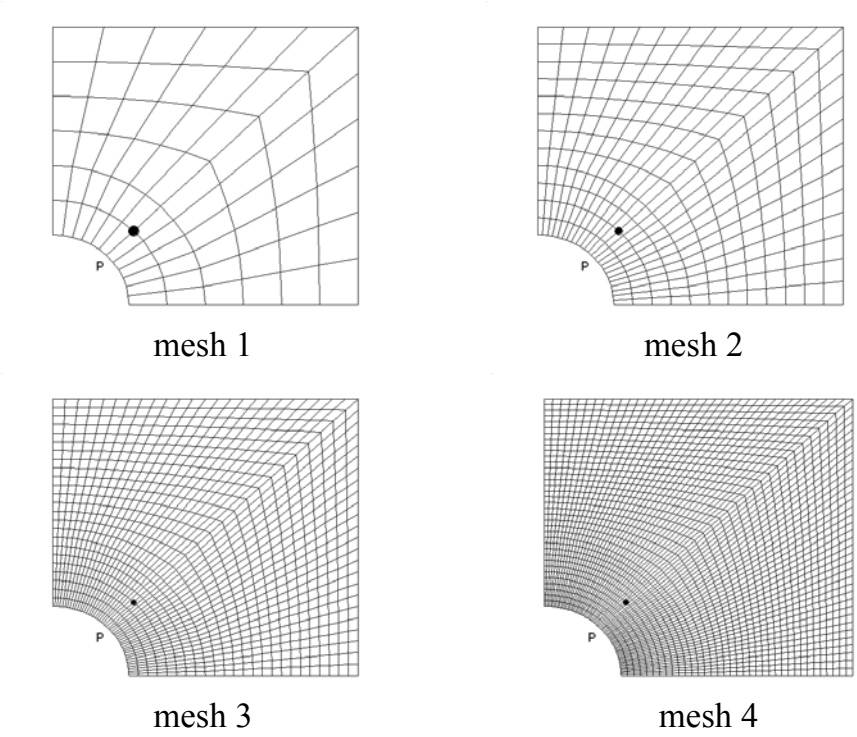


Figure 4.8: Four different meshes of 4-node quadrilateral elements used in analysis (vertex point P locates at (x,y)=(1.0606602, 1.0606602) or (r,θ)=(1.5, 45°))

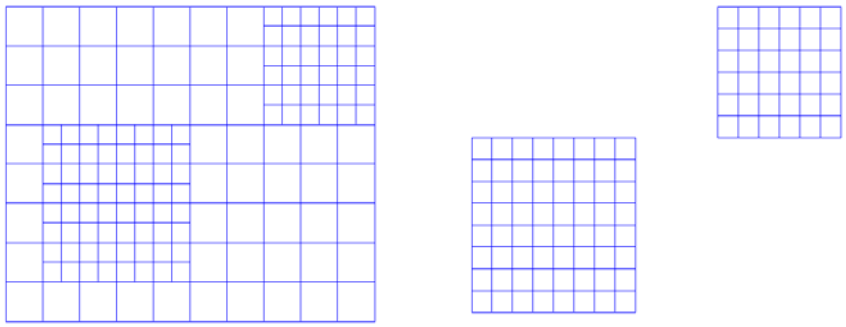
Table 4.1 shows a comparison between the recovered stresses and the exact stresses at vertex point **P**. The percent errors stand for the percent error of the recovered stresses with respect to the exact stresses. It is observed that the *rotated SPR* procedure overcomes the rank deficiency problem and achieves accurate recovered solutions.

Table 4.1: Recovered stresses computed at vertex point P using the *rotated SPR* procedure (percent error shown in parentheses; exact solution shown for comparison)

	σ_x^*	σ_y^*	τ_{xy}^*
Mesh 1	1.0667 *(7.09%)*	-0.0655 *(55.79%)*	-0.2355 *(5.98%)*
Mesh 2	1.1304 *(1.55%)*	-0.1303 *(12.05%)*	-0.2252 *(1.34%)*
Mesh 3	1.1437 *(0.39%)*	-0.1437 *(3.00%)*	-0.2229 *(0.31%)*
Mesh 4	1.1462 *(0.17%)*	-0.1462 *(1.31%)*	-0.2225 *(0.13%)*
Exact Solution	1.1481481	-0.1481481	-0.22222229

4.4 Layer-by-Layer SPR scheme

The SPR technique and the Z^2 error estimator have been widely applied on the conventional finite element meshes used to model a wide variety of physical phenomena. In order to utilize the SPR technique and the Z^2 error estimator in conjunction with the s-refinement, the SPR technique must be applied on the composite mesh generated by mesh superposition, which has not been reported in the literature. The composite mesh poses difficulty in implementing the SRP procedure because it is hard to identify element patches for some overlay nodes. A typical two-layer composite mesh is shown in Figure 4.9(a). The boundary conditions are omitted for the sake of simplicity. It is hard to specify element patches for the overlay nodes located on the boundary of the overlay mesh because the elements surrounding these overlay nodes consist of a mixture of global elements and overlay elements. A *Layer-by-Layer SPR* scheme is thus developed to overcome the difficulty.



(a) (b)

Figure 4.9: a) A typical two-layer composite mesh; b) the corresponding overlay meshes

First, the total displacements for all nodes in the composite mesh are calculated; and then the *Layer-by-Layer SPR* scheme depicted in Figure 4.10 is implemented to obtain the recovered nodal stresses for all nodes in the composite mesh.

Loop over mesh levels (starting from the highest level mesh)
  Loop over all the elements on the current mesh level **M**
    Loop over all the nodes in the current element **E**

      If the recovered stresses has not been calculated for the current node **N**
        1. Calculate the recovered nodal stresses for the current node **N** only using the elements on the current mesh level **M** by the standard SPR procedure.

        2. Copy the computed recovered nodal stresses of the current node **N** to all the nodes on lower mesh levels which share the same spatial location as the current node **N** as their recovered nodal stresses.
      End

    End loop over nodes
  End loop over elements
End loop over mesh levels

Figure 4.10: Flowchart of the Layer-by-Layer SPR scheme

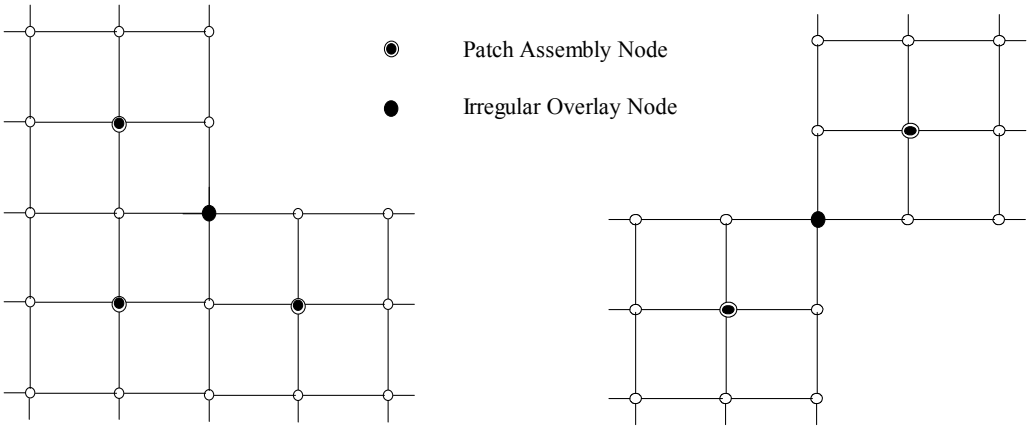
Note that in the scheme, while calculating recovered nodal stresses for a specific node ***N*** on a specific mesh level ***M***, we use only the elements on the mesh level ***M*** to form the element patch. When the current node ***N*** is on the higher *mesh level*, or *mesh level 1* as in Figure 4.9(a), the standard SPR procedure is conducted for node ***N*** on the overlay mesh (*mesh level 1*) as shown in Figure 4.9(b) though the overlay mesh (*mesh level 1*) may not cover the whole problem domain and thus appears discrete. After obtaining the recovered nodal stresses $\bar{\boldsymbol{\sigma}}^*$ for node ***N*** on the higher *mesh level*, the same recovered nodal stresses $\bar{\boldsymbol{\sigma}}^*$ are being assigned to all the nodes in the composite mesh, which are on

lower mesh levels than mesh level ***M*** and share the same spatial location as the node ***N***. The rationale is based on the fact that the recovered nodal stresses of the node ***N*** is computed by the SPR technique on higher level of mesh discretization (mesh level ***M***), which is necessarily more accurate than the SPR solution on lower level of mesh discretizations (mesh levels lower than mesh level ***M***). Due to the structured mesh superposition, it is guaranteed that the scheme provides recovered nodal stresses for any node in the composite mesh.

The structured mesh superposition may generate *irregular overlay nodes* which pose difficulty in implementing the *Layer-by-Layer SPR* scheme depicted in Figure 4.10. A *regular overlay node* is an overlay node which is surrounded by four neighboring overlay elements. Obviously, the four neighboring elements form a valid element patch for the regular overlay node. In contrast, an *irregular overlay node* is an overlay node for which a valid element patch of four overlay elements can not be formed. Several typical irregular overlay nodes labeled with ● are shown in Figure 4.11. If the irregular overlay nodes appear, the *average scheme* [7] is utilized. The recovered values for the irregular overlay nodes are taken to be the average of the recovered values from all the relevant patches identified by the patch assembly nodes in Figure 4.11.

After recovered nodal stresses are calculated for all nodes in the composite mesh, the Z^2 error estimator is calculated. The concept of *top element* in the composite mesh is

introduced, which is an element which is not superimposed by any higher-level overlay elements. Intuitively, *top element*s are those visible elements on a top view of the composite mesh. Although all top elements may not necessarily be on same mesh level, they obviously form a seamless and non-overlapping mesh that fully covers the problem domain Ω. The mesh formed by all top elements is referred to as *top mesh* in this study. In Figure 4.9(a), the *top elements* are all the elements on *mesh level 1* and a portion of elements on *mesh level 0* which are not covered by elements on *mesh level 1*. In computing the estimated error $\|\mathbf{e}\|_E$ on the problem domain Ω as in section 4.2, the integration is conducted only on the *top mesh* in the composite mesh.



(a) Irregular Overlay Node type I

(b) Irregular Overlay Node type II

Figure 4.11: Handling of the irregular overlay nodes in the composite mesh with the Layer-by-Layer SPR scheme

The effectiveness of the *Layer-by-Layer SPR* scheme will be demonstrated via a series of numerical examples shown later in Chapter 5, Chapter 6 and Chapter 7.

Chapter 5: STRUCTURED S-ADAPTIVITY IN 2-D/3-D LINEAR STATICS

Adaptive methods that use finite element mesh superposition as the refinement scheme are referred to as *s-adaptive* methods and have been successfully applied to 2-D linear elastostatic problems [15-26]. In this chapter, the details of the s-adaptive method are presented in section 5.1 and then applied in section 5.2 to the classic 2-D L-shaped domain problem with stress concentrations to demonstrate its superior performances. More importantly, the focus and originality of this chapter is on the extension of the s-adaptive method to 3-D adaptive analysis in section 5.3, and its applications in the 3-D problems in section 5.4.

5.1 S-adaptive Method

For static problems, an adaptive analysis attempts to find a near-optimal mesh discretization which achieves a user-specified tolerance η_s^{TOL} on the spatial error with the minimal amount of computational resources possible. This goal is typically achieved by locally adjusting the density of the mesh based on local error estimations. These local refinements continue to be applied until the estimated relative global error η_s meets the user-specified tolerance as in equation (5.1).

$$\bar{\eta}_s \le \eta_s^{TOL} \tag{5.1}$$

The general procedure of an adaptive analysis in linear static problems is as follows. First, the problem is solved with a coarse mesh and the global spatial error is evaluated. If the accuracy requirement in (5.1) is not met, the solution is rejected. Next, the mesh is refined based on local error estimation within each individual element, and the problem is solved again on the new (refined) mesh. This process is repeated until the required global accuracy is achieved. In the s-adaptive method, consistent with the definition of the global spatial error in the energy norm, the element error for *top element i* in the energy norm is defined as

$$e_i = \int_{\Omega_i^e} (\boldsymbol{\sigma}^* - \boldsymbol{\sigma})^{\mathrm{T}} \mathbf{D}^{-1} (\boldsymbol{\sigma}^* - \boldsymbol{\sigma}) d\Omega_i^e \tag{5.2}$$

where Ω_i^e is the domain of top element *i*, while $\boldsymbol{\sigma}$ and $\boldsymbol{\sigma}^*$ are the raw finite element stress distributions and the spatially postprocessed (improved) stress distributions, respectively.

The optimal mesh hypothesis [45, 46] asserts that an optimal mesh not only meets the global accuracy requirement, but also results in a uniform spatial error density throughout the problem domain. Thus, for any particular element in the composite mesh, regardless of the mesh level on which it appears, its element error threshold should be based on an allowable error density, i.e. the element error threshold is proportional to the size of the element and inversely proportional to the size of the computational domain. The element error threshold for element *i* is defined as

$$\bar{e}_i = \frac{\mathrm{A}_i}{\mathrm{A}_\Omega}\theta_s\left(\eta_s^{TOL}\left\|\mathbf{u}^h\right\|_E\right)^2 \tag{5.3}$$

where A_i is the area of element and A_Ω is the area of the problem domain Ω. $\theta_s < 1.0$ is a safety factor which increases the likelihood that the new mesh will satisfy the specified error tolerance.

If $e_i > \bar{e}_i$ for a *top element i*, the offending element will be superimposed with a $\mathrm{d_h}\times\mathrm{d_v}$ patch of overlay elements to locally enhance the composite mesh. It is briefly proved in (5.4) that the satisfaction of the accuracy requirement is guaranteed if every top element has an element error e_i less than its error threshold $\bar{e}_i$.

$$\begin{aligned}
&\left\|\mathbf{e}_\mathrm{s}\right\|_\mathrm{E} = \left[\int_\Omega (\boldsymbol{\sigma}^* - \boldsymbol{\sigma})^\mathrm{T}\mathbf{D}^{-1}(\boldsymbol{\sigma}^* - \boldsymbol{\sigma})d\Omega\right]^{1/2} = \left[\sum_i \int_{\Omega_i^e} (\boldsymbol{\sigma}^* - \boldsymbol{\sigma})^\mathrm{T}\mathbf{D}^{-1}(\boldsymbol{\sigma}^* - \boldsymbol{\sigma})d\Omega_i^e\right]^{1/2} = \left[\sum_i e_i\right]^{1/2} \\
&\le \left[\sum_i \bar{e}_i\right]^{1/2} = \left[\frac{\sum_i \mathrm{A}_i}{\mathrm{A}_\Omega}\theta_s\left(\eta_s^{TOL}\left\|\mathbf{u}^h\right\|_E\right)^2\right]^{1/2} = \left[\frac{\mathrm{A}_\Omega}{\mathrm{A}_\Omega}\theta_s\left(\eta_s^{TOL}\left\|\mathbf{u}^h\right\|_E\right)^2\right]^{1/2} = \sqrt{\theta_s}\left(\left\|\mathbf{u}^h\right\|_E\right)\eta_s^{TOL}
\end{aligned} \tag{5.4}$$

5.2 2-D Numerical Example

The performance characteristics of the structured s-adaptive method are demonstrated in a 2-D linear static problem which features local stress concentrations. Only spatial discretization errors are involved in the static problems.

5.2.1 2-D L-shaped Domain Problem

The classic 2-D L-shaped domain problem features a singularity in the stress field caused by the presence of a re-entrant corner. The geometry, material properties, constraints and external loading are given in Figure 5.1. A state of plane stress is assumed and 4-node quadrilateral elements are used to discretize the problem. The spatial error tolerance is set at η_s^{TOL} =4.5%. The overlay mesh parameters are chosen as $d_h = d_v = 2$, i.e. a 2×2 overlay mesh is superimposed to provide additional refinement within any existing *top element* that fails to achieve the specified error density tolerance. This provides additional local refinement needed to resolve the solution in regions of high solution gradients. Additional levels of overlay meshes are superimposed locally on the existing top elements of the composite mesh until the desired spatial error tolerance is met. For this particular problem, a total of five levels of overlay mesh are required to achieve the specified error tolerance.

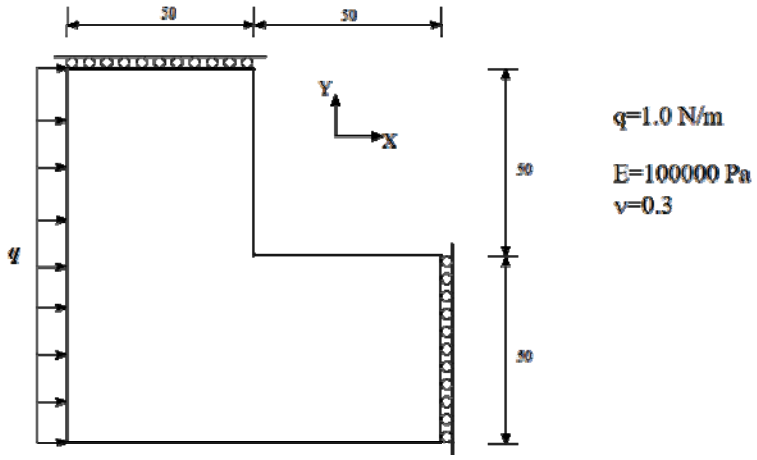


Figure 5.1: A 2-D, L-shaped domain subject to distributed static load

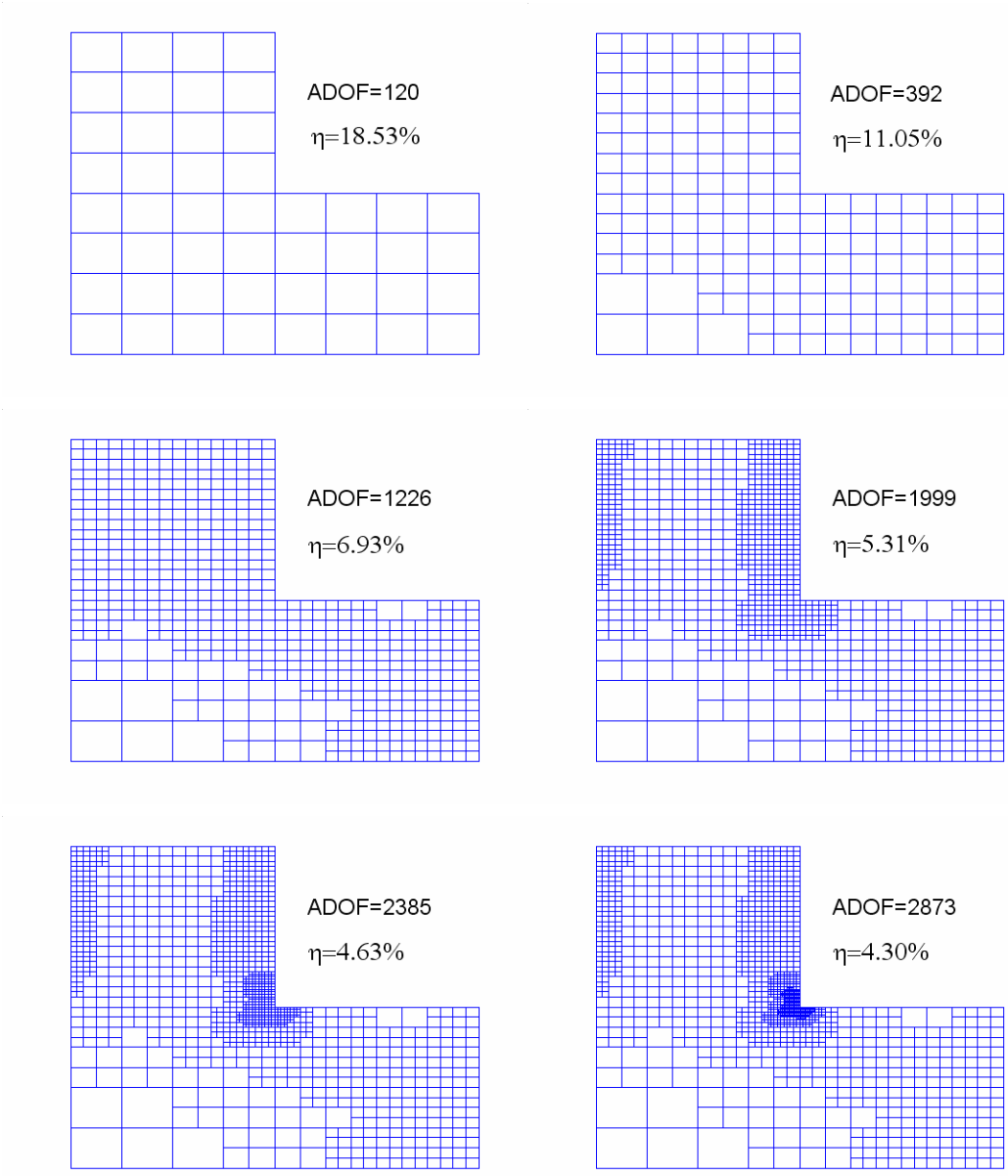


Figure 5.2: Series of composite meshes and estimated relative errors by the structured s-adaptive analysis. ADOF denotes the number of active degrees of freedom

Figure 5.2 shows the series of composite meshes and the corresponding estimated spatial errors that are generated by the s-adaptive procedure. As seen in Figure 5.2, the first three levels of overlay meshes are distributed over a significant portion of the computational domain, while the two highest levels of overlay meshes are needed only in

the immediate vicinity of the stress singularity at the re-entrant corner. For comparison, the L-shaped domain problem is also solved using a series of uniformly refined conventional meshes. Figure 5.3 shows the relative spatial error achieved by both analyses as functions of the number of active degrees of freedom. The structured s-adaptive analysis is clearly able to achieve the same low spatial errors using fewer degrees of freedom than the uniform refined analysis.

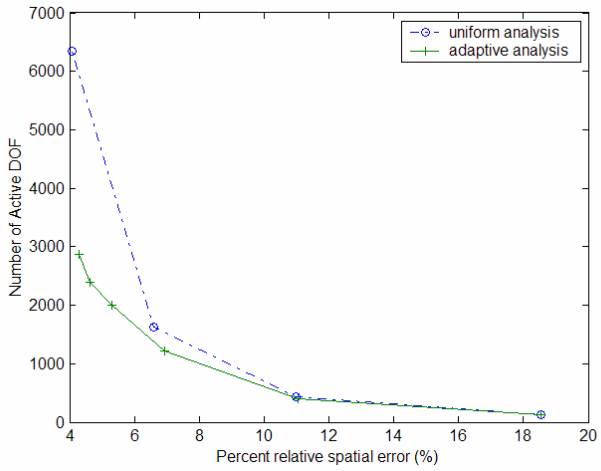


Figure 5.3: Relationship between the number of active DOF and the percent relative error achieved by the s-adaptive composite meshes and the uniform refined meshes

5.3 Structured S-adaptivity in 3-D Problems

As mentioned in Chapter 1, the s-refinement is able to overcome many difficulties associated with traditional h-refinement. In the application of the s-adaptive methodology in 2-D linear elastic static problems, we observed the simplicity and universality in producing the required refinement by superimposing overlay meshes on the critical areas.

More importantly, the simplicity and universality features of mesh generation are preserved in the extension of the s-adaptive methodology from 2-D to 3-D problems. Therefore, we expect the s-adaptive methodology can be easily extended from 2-D to more complex and computational demanding 3-D without a disproportionate increase in programming complexity and solution expense as occurred in the complete remeshing type h-adaptive methodology.

Although the mesh superposition technique is extended to 3-D problems, all the characteristics of the methodology are exactly the same as in 2-D problems. The local displacement fields still serve as relative displacements or incremental enhancement to the global displacement fields, thus resulting in the same hierarchical structure in the system matrices and vectors. The only difference is due to the increase of the geometry complexity from 2-D to 3-D. The geometric entities in 2-D are upgraded in 3-D in the sense of areas upgraded to volumes, lines upgraded to faces and points upgraded to lines. Although these upgrades of geometric entities pose more difficulties in 3-D problems, the essence of the methodology of mesh superposition technique is not altered. The standard derivations will not be repeated, however, several other key issues in the extension from 2-D to 3-D is being briefly presented. The discussion and demonstration of the extension of s-adaptivity to the 3-D case will be limited to the use of the 8-node, 3-D linear hexahedral element since this is the 3-D analog of the 2-D linear quadrilateral element used earlier in the discussion of 2-D problems. It should be emphasized that the use of

simple linear 2-D and 3-D elements is not a restriction of the s-adaptive method, rather the simple linear elements are chosen because they possess certain advantages for modeling transient behavior of localized phenomena like wave propagation, plasticity and damage.

5.3.1 SPR Technique in 3-D

In the previous s-adaptive analysis of 2-D problem, the superconvergent patch recovery (SPR) technique was used to improve the accuracy of the stress field that comes directly from the finite element solution. The extension of s-adaptive method from 2-D to 3-D necessitates the extension of the SPR postprocessing technique from 2-D to 3-D as described by Choi and Lee [14]. For the 8-node, 3-D hexahedral element, the improved, continuous stress field on an element patch is assumed in the form of an eight term Legendre polynomial $\mathbf{P} = [1, x, y, z, xy, xz, yz, xyz]$.

Figure 5.4(a) shows a typical 8-node brick element and its superconvergent sampling point labeled with ▲ at its reduced Gaussian integration point. Since there are eight terms in the Legendre polynomials **P**, a typical superconvergent patch is assembled by eight brick elements around the assembly node in the center which is shown in Figure 5.4(b). Figure 5.4(c) and Figure 5.4(d) show the patches on the boundary and the patch at the corner, respectively.

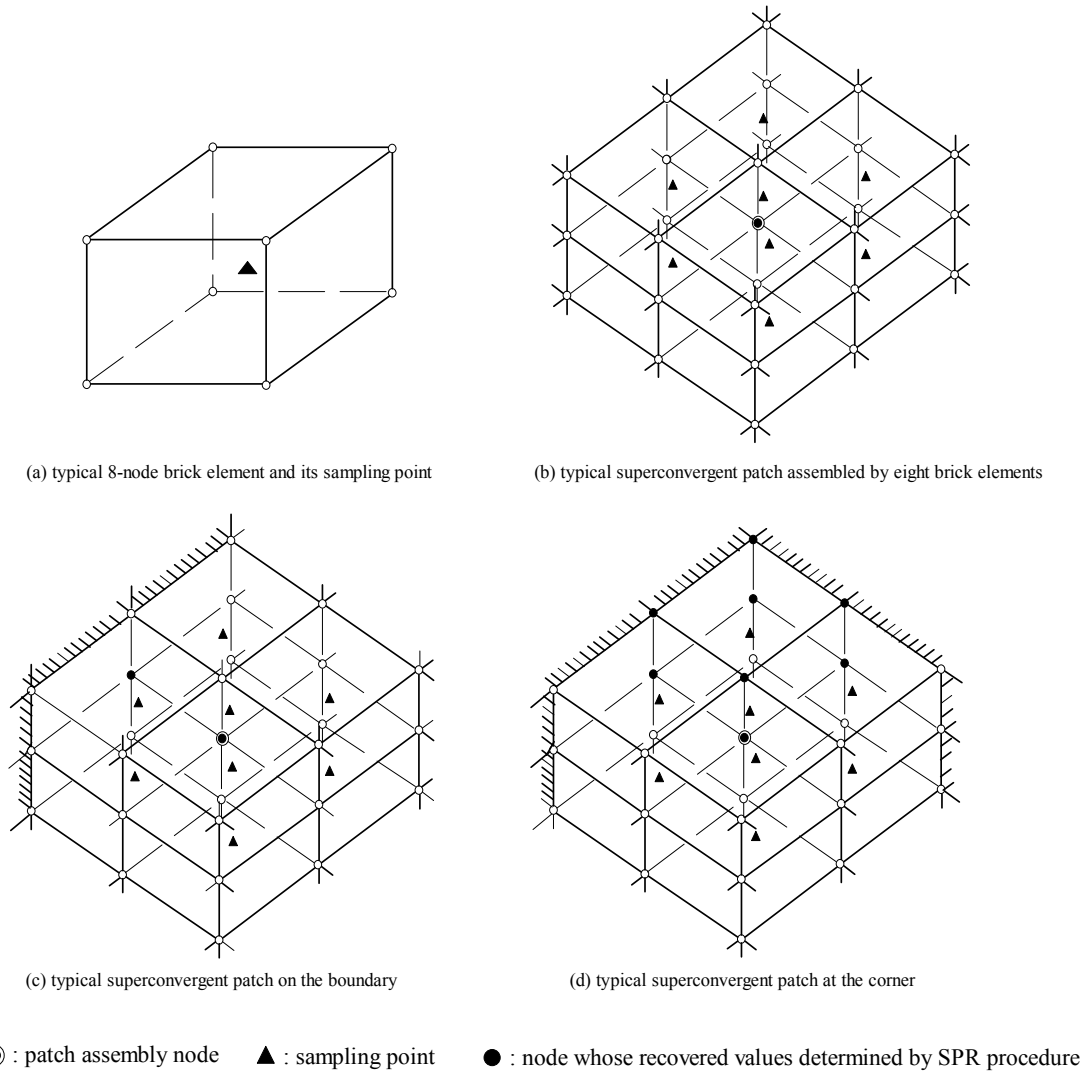


Figure 5.4: Typical 3-D superconvergent patch for 8-node brick element

5.3.2 S-refinement Scheme in 3-D

The s-refinement scheme in 3-D is very similar to the s-refinement scheme used in 2-D. Here in 3-D, we borrow the concept of the *top element* as developed earlier for 2-D. A *top element* is any element in the composite mesh which is not superimposed by any overlay element. First, the element errors and the corresponding element error thresholds of all

top elements are calculated using equation (5.2) and equation (5.3), respectively. If the error of a top element exceeds its threshold, the offending top element is effectively refined by superimposing a structured higher lever of overlay patch constructed by *overlay mesh parameters* d_l , d_w and d_h, which represent the number of elements in the overlay patch generated in the directions of length, width and height, respectively. The s-refinement scheme is repeated iteratively until the final composite mesh is able to provide a solution which meets the accuracy requirement in (5.1).

5.3.3 Layer-by-Layer SPR Scheme in 3-D

A new *Layer-by-Layer SPR* scheme is developed for 2-D problems in section 4.4 to implement the standard SPR technique in the composite mesh generated by the structured mesh superposition. In this section, the same *Layer-by-Layer SPR* scheme is extended to 3-D problems.

In Figure 4.4 of section 4.3, we present several irregular overlay nodes in 2-D. Although the irregular overlay nodes in 3-D are more complex due to the geometrical complexity in 3-D, the strategy of the *Layer-by-Layer SPR* scheme remains unchanged. For the regular overlay nodes which are surrounded by eight neighboring overlay elements, the recovered values are computed by the standard SPR procedure using an element patch formed the eight neighboring elements. However, for the irregular overlay nodes which

are contained in less than eight overlay elements, the *average scheme* is utilized. The recovered values are taken to be the average of the recovered values from several relevant patches. Figure 5.5 shows a typical irregular overlay node which is labeled with ● and connects two overlay patches which are identified by the assembly nodes locating in the center of the patches.

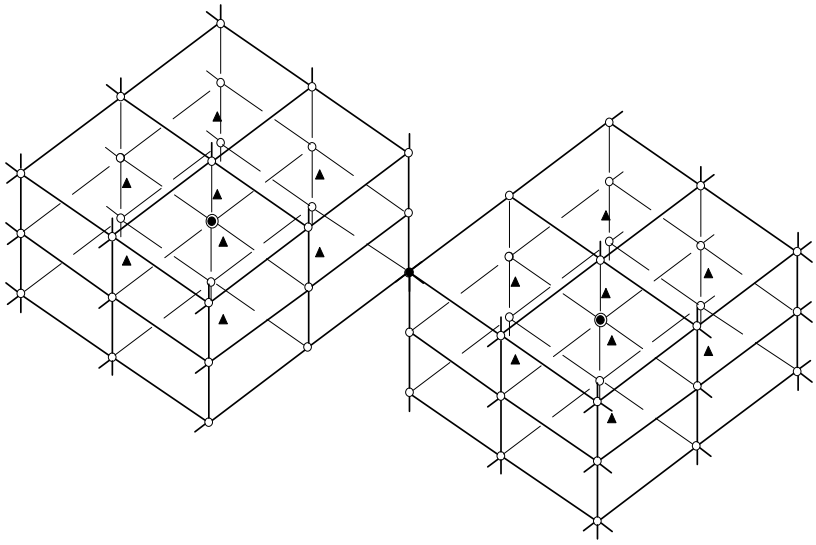


Figure 5.5: Handling of the irregular overlay node in composite mesh with SPR

In Figure 5.5, the connecting overlay node is only contained in two overlay elements, which are not enough to form an eight-element patch required by the standard SPR. Therefore, the recovered values of the irregular overlay node are taken to be the average of the recovered values from the two standard SPR patches. It is noteworthy that the configuration shown in Figure 5.5 is not the only form that the irregular overlay nodes may take. Although the geometric complexity in 3-D causes many other configurations, the *average scheme* is similarly applicable.

5.4 3-D Numerical Examples

Two simple linear static examples problems in 3-D are chosen to demonstrate the effectiveness of the mesh superposition technique for 3-D problems. Only spatial discretization errors are involved in these static problems. Both problems involve 3-D cantilevered beams subjected to uniform distributed transverse loading; however, the second problem involves an incomplete cantilevered condition (i.e. a flaw in the boundary support conditions). Thus, the form of the solution is much more complex in the second example. The 3-D, 8-node, linear hexahedral element is used for both examples.

5.4.1 Fully Clamped 3-D Cantilever Problem

Figure 5.6(a) shows a 3-D cantilever with left end fully clamped. The cantilever is subjected to a uniform distributed load on the top surface. The relative spatial error tolerance is chosen as 10%. The overlay mesh parameters are chosen as $d_l = d_w = d_h = 2$. Figure 5.6(b) shows the 2×2×10 global mesh (i.e. the level-0 mesh) that is used to initiate the s-adaptive solution. Figure 5.6c shows the composite mesh after superimposing level-1 overlay patches over those global elements that failed to meet the error tolerance. However, the composite mesh shown in Figure 5.6c still contains many top elements that exceed the error tolerance. Figure 5.6d shows the composite mesh after superimposing

level-2 overlay patches. The composite mesh shown in Figure 5.6d proved to be capable of meeting the user-specified error tolerance of 10%. It is observed that the overlay degrees of freedom are concentrated toward the critical area near the clamped end where the stress gradients are more rapidly changing. The 3-D figures are generated in FEAP 7.1 [78]. The numerical test results are presented in Table 5.1 which shows the reduction in error achieved by each successive composite mesh used in the s-adaptive solution.

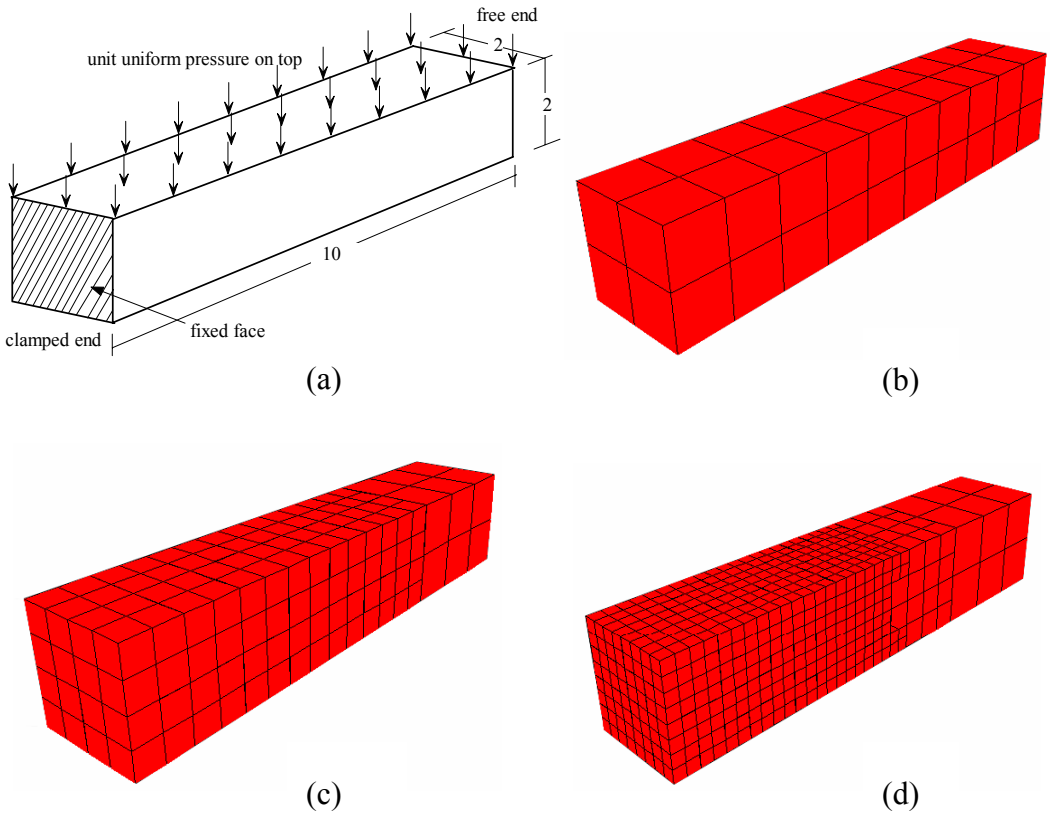


Figure 5.6: (a) Problem definition including the geometry, constraints and loads; (b) global mesh; (c)-(d) adaptively refined composite meshes

Table 5.1: Mesh refinement results for fully clamped cantilever

Layer of Composite Mesh	Number of Active Elements	Number of Active DOF	$\|u\|$	Relative Percent Error
1	40	270	3.0445	32.05
2	236	1083	3.1891	17.98
3	1412	5211	3.2318	9.97

For comparison, the same problem is also solved using a series of uniformly refined, non-adaptive meshes. Figure 5.7 shows the relative spatial error achieved by both methods as functions of the number of active degrees of freedom. The structured mesh superposition method is clearly able to achieve higher accuracy using fewer degrees of freedom than the uniform refined meshes.

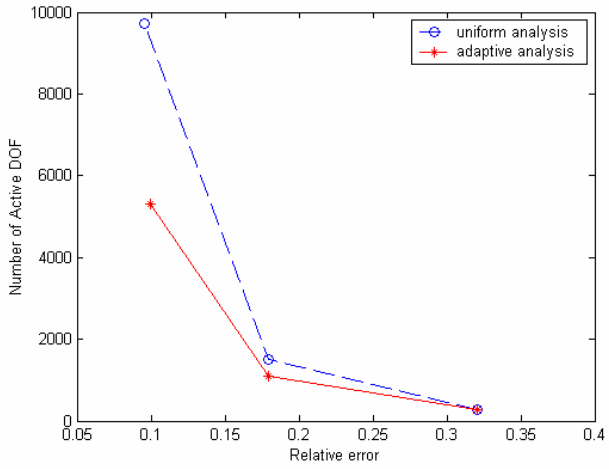


Figure 5.7: Relationship between the number of active DOF and the relative error achieved by the s-adaptive composite meshes and the uniform refined meshes

5.4.2 Partially Clamped 3-D Cantilever Problem

Here, we consider a 3-D cantilever subjected to a uniform distributed transverse load on the top surface; however, this problem differs from the previous example in that the cantilevered support is not applied to the entire left-hand boundary. Instead, the top-right quadrant of the cantilevered surface is left free, thus simulating a square crack in the cantilevered support. Figure 5.8(a) shows the problem geometry, material properties, boundary constraints and external loading. Because of the abrupt change in the boundary conditions on the clamped end, a stress concentration is expected to occur along the 'L-shaped' line that separates the clamped region from the non-clamped region.

Figure 5.8(b) shows the initial global mesh (the level-0 mesh) used to initiate an s-adaptive solution with a spatial error tolerance chosen as η=15% and overlay mesh parameters chosen as $d_l = d_w = d_h = 2$. Figures 5.8(b) through 5.8(e) show the composite mesh after applying the necessary level-1, level-2, level-3 and level-4 overlay patches in an effort to satisfy the user-specified spatial error tolerance. In particular, Figure 5.8(e) clearly shows the concentration of overlay degrees of freedom along the L-shaped line that represents the crack front. The 3-D figures are generated in FEAP 7.1 [78]. The numerical test results are presented in Table 5.2 which shows the spatial error achieved by each composite mesh used in the adaptive solution process.

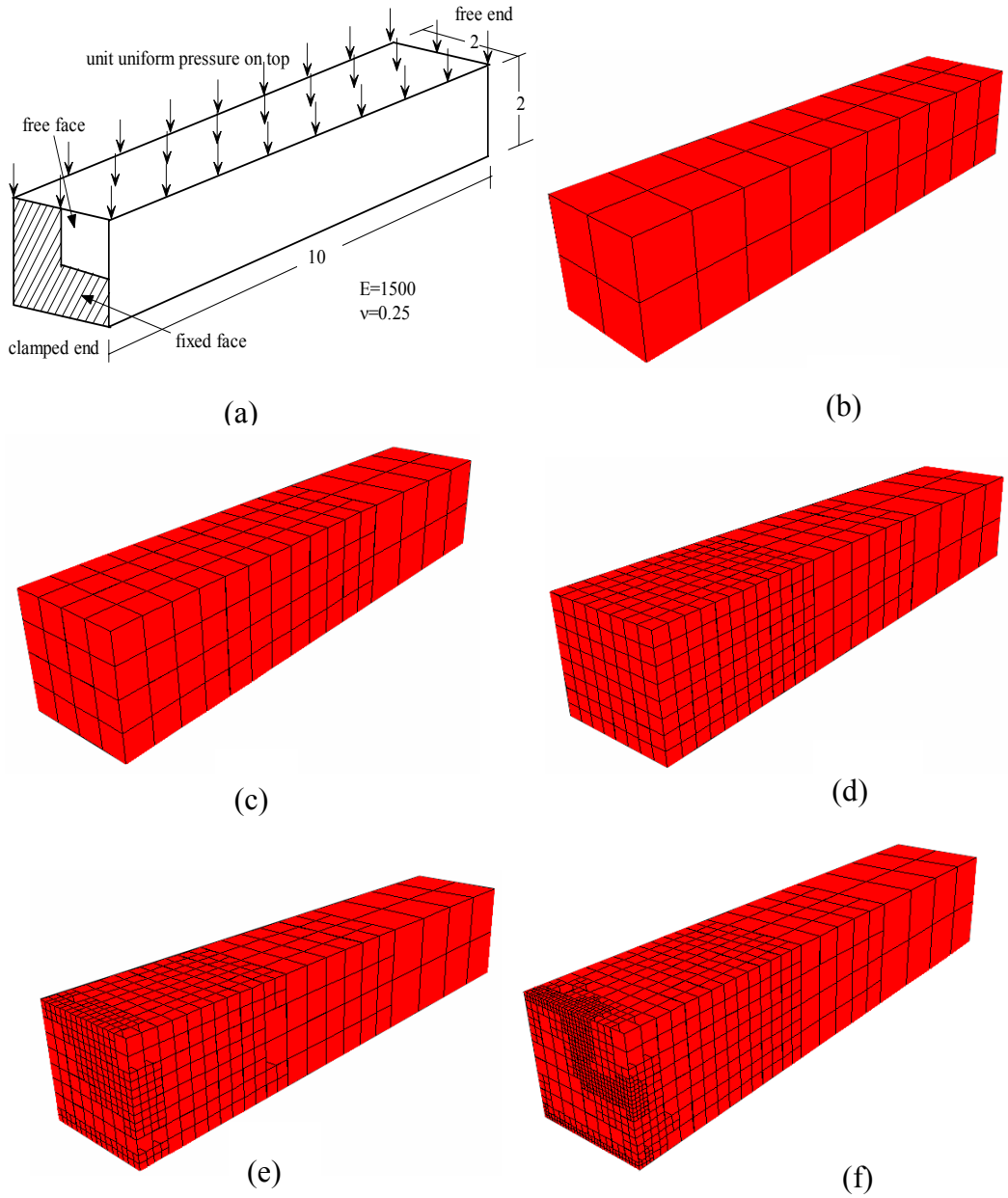


Figure 5.8: (a) Problem definition; (b) global mesh; (c)-(f) adaptively refined meshes

Table 5.2: Mesh refinement results for partially clamped cantilever

Layers of Composite Mesh	Number of Active Elements	Number of Active DOF	$\|u\|$	Relative Percent Error
1	40	273	3.1880	37.28
2	208	972	3.3765	24.34
3	880	3306	3.4461	16.94
4	1265	4131	3.4637	15.54
5	2875	8130	3.4769	12.67

For comparison, this problem is also solved using a series of uniformly refined, non-adaptive meshes. Figure 5.9 shows the relative spatial error achieved by both s-adaptive and non-adaptive methods as functions of the number of active degrees of freedom. The structured mesh superposition method is clearly able to achieve higher accuracy using fewer degrees of freedom than the uniform refined meshes.

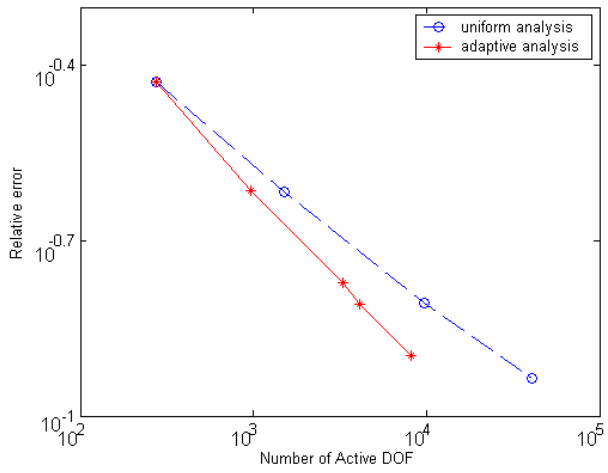


Figure 5.9: Relationship between the number of active DOF and the percent relative error achieved by the s-adaptive composite meshes and the uniform refined meshes

Chapter 6: STRUCTURED S-ADAPTIVITY IN LINEAR ELASTODYNAMICS

In this chapter, an s-adaptive procedure is developed for 2-D linear elastodynamics. The resulting procedure utilizes both temporal adaptivity and spatial adaptivity so that it automatically updates both the spatial mesh and the time step to achieve high efficiency while maintaining the required accuracy. The spatial adaptivity is achieved using the *s-adaptive* methodology presented in Chapter 5. The details of the s-adaptive procedure are presented in section 6.1, followed by two 2-D numerical examples in section 6.2.

6.1 S-adaptive Procedure

For transient problems, an adaptive analysis should attempt to maintain an adequate yet efficient level of spatial and temporal discretization over the entire space and time domain of the analysis so that the relative spatial and temporal errors are kept within user-specified tolerances, η_s^{TOL} and η_t^{TOL} respectively, i.e., we require both

$$\bar{\eta}_t \le \eta_t^{TOL} \qquad \text{and} \qquad \bar{\eta}_s \le \eta_s^{TOL} \tag{6.1a,b}$$

After accepting solutions at time t_n with a particular spatial mesh and a particular time step size Δt_n, we then obtain an initial solution at time $t_{n+1} = t_n + \Delta t_n$ using this same mesh. However, before accepting the new solution, it must be first subjected to the following *temporal adaptivity* and *spatial adaptivity* processes.

6.1.1 Temporal Adaptivity

Temporal adaptivity starts with temporal error estimation. In many studies [41, 43], the error per time step (i.e. the local temporal error) is used for the control of temporal error. In a given time step, if the requirement (6.1a) is not satisfied, the solution is rejected and the time step size is refined until the required accuracy is reached. Since the convergence rate of temporal error in a local time step is $O(\Delta t^3)$, the new step size may be predicted as:

$$\Delta t^{new} = \left(\frac{\theta_t \eta_t^{TOL}}{\overline{\eta}_t} \right)^{1/3} \Delta t \tag{6.2}$$

where Δt is the current time step size and $\theta_t < 1.0$ is a factor of safety introduced to ensure the acceptance of the newly predicted time step.

If the estimated temporal error is less than the tolerance, then the current time step size is deemed adequate. In the case where the estimated temporal error happens to be much less than the tolerance, the time step size may be increased according to equation (6.2). However, to limit the number of time step size changes, increases to the time step size are only performed when the condition $\overline{\eta}_t < \beta_t \eta_t^{TOL}$ is met for K_t successive time steps. Here, both K_t and β_t are user-specified parameters, where K_t is an integer and $0<\beta_t<1$ serves as a factor of safety introduced to ensure the acceptance of the newly increased time step size.

6.1.2 Spatial Adaptivity

The spatial adaptivity starts with spatial error estimation. If the requirement (6.1b) is not satisfied, the current solution is rejected and the spatial mesh should be adapted. Also, in the case where the estimated spatial error is much less than the tolerance error η_s^{TOL}, a coarser mesh may be capable of providing the specified level of accuracy with increased computational efficiency. In this case, the mesh adaptation is triggered when the condition $\overline{\eta}_s < \beta_s \eta_s^{TOL}$ has been met for K_s consecutive time steps, where β_s and K_s are user-defined parameters similar to β_t and K_t, respectively. In summary, mesh adaptation or mesh change can be triggered by two different situations: 1) the current mesh is not adequate to meet the specified spatial error tolerance and; 2) the current mesh contains significantly more refinement than is needed to meet the specified spatial error tolerance.

In the event that a mesh change is indicated, the necessary mesh changes are accomplished by locally comparing the error e_i in each top element i with the user-specified element error threshold $\overline{e}_i$ for top element i. The mesh changes can simultaneously involve *element refining* in certain parts of the mesh and *element coarsening* in other parts of the mesh. The element error e_i and the element error threshold $\overline{e}_i$ for top element i are defined exactly the same as for the earlier static case.

$$e_i = \int_{\Omega_i^e} (\boldsymbol{\sigma}^* - \underline{\boldsymbol{\sigma}}^*)^{\mathrm{T}} \mathbf{D}^{-1} (\boldsymbol{\sigma}^* - \underline{\boldsymbol{\sigma}}^*) d\Omega_i^e \tag{6.3a}$$

$$\bar{e}_i = \frac{\mathrm{A}_i}{\mathrm{A}_\Omega} \theta_s \left(\eta_s^{TOL} \left\| \mathbf{u}^h \right\|_E \right)^2 \tag{6.3b}$$

In equation (6.3a), Ω_i^e is the domain of top element *i* and $\underline{\boldsymbol{\sigma}}^*$ and $\boldsymbol{\sigma}^*$ are the improved stress distributions that are computed during temporal and spatial postprocessing respectively; A_i is the area of top element *i*, A_Ω is the area of the computational domain Ω, and $0 < \theta_s < 1.0$ is a safety factor.

ELEMENT REFINING: If $e_i > \bar{e}_i$, then the offending top element *i* is superimposed with a single overlay patch of $d_h \times d_v$ overlay elements (or sibling elements). Here, the phrase 'sibling elements' refers to the overlay elements of a single overlay patch that is superimposed on a single lower level element. These sibling elements are considered as the 'children elements' of the lower level element on which they are superimposed. In this case, the lower level element is considered to be the 'parent element'.

ELEMENT COARSENING: This operation occurs whenever *all* of the sibling elements of a single top-level overlay patch exhibit element error e_i that is significantly less than their corresponding element error threshold $\bar{e}_i$. This condition is expressed as $e_i \leq \phi \bar{e}_i$ for $i=1,2,\ldots,d_h \times d_v$, where $0<\phi<1$ is a user-specified parameter to control the coarsening of elements and the integer $d_h \times d_v$ is the number of overlay elements in the overlay patch.

If this condition is satisfied for a single top level overlay patch, then the entire patch of $d_h \times d_v$ overlay elements is removed from the composite mesh.

After element refining and element coarsening are applied to all top elements in the old composite mesh, a new composite mesh is generated, where unnecessary local refinement is removed and the necessary local refinement is created simultaneously.

6.1.3 Data Transfer Scheme

Any changes made to the composite mesh require the solution from the old composite mesh to be transferred/interpolated to the new composite mesh. In a linear elastodynamic analysis, the solution data that must be transferred includes only *node-wise* variables such as the displacements, velocities and accelerations.

In traditional h-adaptive procedures that involve extensive reformulation of the entire mesh (e.g. [45]), the process of interpolating the solution from one mesh to another is problematic for several reasons. First, the process is quite cumbersome and adds significantly to the overall computational expense. Second, the process requires the availability of a complex mesh generator which adds significantly to the overall development effort of the adaptive code. Third, the process of mesh reformulation makes it unlikely that the old mesh and the new mesh will share many common nodes, consequently the solution transfer process generally introduces a considerable amount of

additional spatial error into the transient solution at the current time t (i.e. at the time of mesh reformulation). In a transient solution, this additional spatial error is especially troublesome since the newly transferred solution serves as the initial conditions for the next time integration step. Some of these disadvantages can be lessened by using an h-adaptive procedure that relies on subdivision and merger of existing elements (e.g. [46]), but this procedure generates many incompatible (or hanging) nodes, each of which requires the imposition of multipoint constraints, thus making the procedure exceedingly difficult for 2-D and 3-D problems.

The currently proposed structured *s-adaptive* procedure circumvents these difficulties in the sense that the mesh refinement process is both simple and fast, multipoint constraints are not required, and interpolation error is non-existent. These advantages all stem from the hierarchical data structure that is used in the formulation. The speed and simplicity of the mesh adaptation process is achieved by restricting the mesh superposition to the structured form which creates overlay mesh patches (at any mesh level) by simply subdividing existing elements on the next lower level. Since each of these overlay patches (on mesh level m) have the same mesh density ($d_h \times d_v$) and are each superimposed on a single element of a compatible mesh at level $m-1$, the overlay patches are necessarily compatible with each other. After superimposing a new overlay mesh, the compatibility and uniqueness requirements of the resulting composite mesh are achieved by simply suppressing the overlay degrees of freedom at certain overlay nodes,

thus avoiding the use of cumbersome multi-point constraints or transition elements. This is possible because the interpolants of the new overlay mesh are hierarchical enhancements to the interpolants of the existing composite mesh. Transferring the solution (i.e. displacements and velocities) to the new composite mesh is particularly simple and does not require explicit interpolation. Any new nodes that are introduced into the composite mesh are used for hierarchical enhancement of the existing solution; therefore, the designated value of the displacements and velocities at these new nodes is simply zero. In this study, the displacements and velocities on the new composite mesh are obtained through direct data transfer from the old composite mesh, while the accelerations of the new composite mesh as in [45].

This data transfer scheme between the old and new composite meshes is illustrated in Figure 6.1 for the simple case of two-level structured mesh superposition.

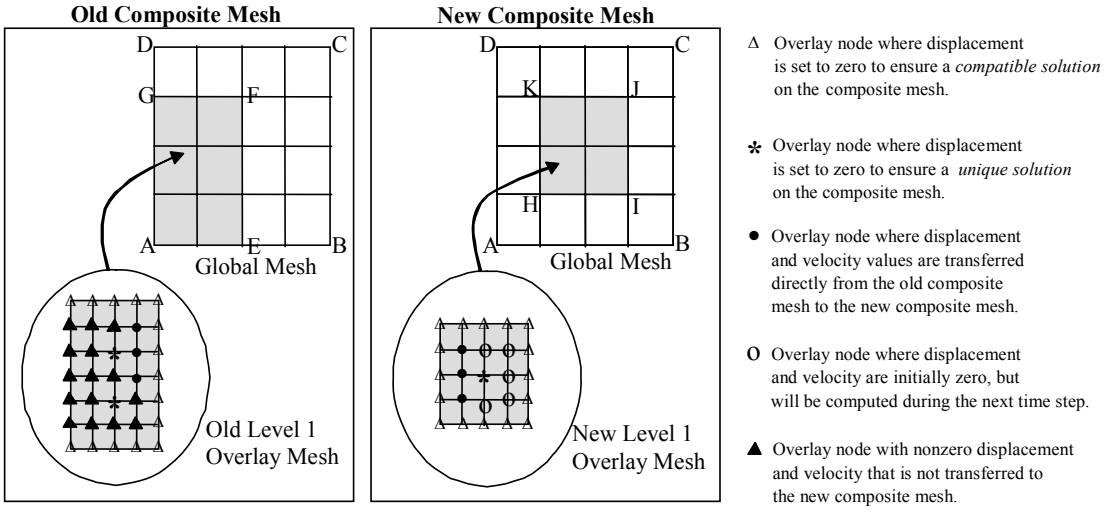


Figure 6.1: Simple data transfer scheme in the s-adaptive procedure

The global mesh (mesh level 0) consists of a 4×4 mesh of 4-node quadrilateral elements, occupying the rectangular domain ABCD. Global essential boundary conditions are assumed along boundary AB. At time t_n, just prior to updating the composite mesh, the overlay mesh (mesh level 1) occupies the shaded rectangular subregion AEFG. In this case, each shaded global element is covered by a 2×2 patch of overlay elements (i.e., $d_h=d_v=2$). In the new composite mesh, the overlay mesh (mesh level 1) occupies the shaded rectangular subregion HIJK, and is formed by covering each of the shaded global elements with a 2×2 patch of overlay elements. To guarantee that the solution on the new composite mesh satisfies the *compatibility* and *uniqueness* requirements, the overlay displacements and velocities at the overlay nodes labeled with the symbols Δ and * are simply suppressed. Next, we directly transfer the displacements and velocities of any remaining overlay nodes that are present and unconstrained in both the old and new overlay meshes (labeled with the symbol ●). Finally, we identify the new nodes that appear in the new overlay mesh but were not present in the old overlay mesh (labeled with the symbol ○). At these nodes, the displacements and velocities are initially set to zero, but will be computed as part of the evolving solution. This data transfer process is exact within any subregions where the mesh density remains constant or increases. However, in the subregions where the composite mesh undergoes coarsening, the overlay displacements at some of the nodes of previous overlay mesh (labeled with symbol ▲) cannot be transferred into the new overlay mesh and are simply discarded. This loss of higher order solution fidelity is a problem that is common to all forms of solution transfer schemes when local mesh coarsening occurs.

However, in the present methodology, this effect is most often negligible since mesh coarsening only occurs when the highest level overlay degrees of freedom are no longer making a significant contribution to the composite solution.

After the solutions of the displacement and velocity are transferred to new composite mesh, the solution of the acceleration is computed for the new mesh by resolving the dynamic equations on the new mesh as in [45].

It should be emphasized that the creation of a new composite mesh that satisfies the specified error tolerance is sometimes an iterative process, i.e., there is no guarantee that the new composite mesh will satisfy the specified spatial error tolerance. In such cases where more than one attempt at a composite mesh creation is required, the processes of error estimation and solution transfer proceed as follows. Error estimation is always computed on the composite mesh that was used to obtain the latest solution. Based on this error estimation, a new composite mesh is then created. Then the solution is transferred to the new composite mesh from the old composite mesh that generated the last acceptable solution. This ensures that a reliable solution is transferred to the new composite mesh, not an unreliable intermediate solution.

6.1.4 Flow Chart

The flow chart of the structured s-adaptive procedure is symbolically depicted in Figure 6.2. In the *Mesh Update Module*, an optimal composite mesh is generated based on the current composite mesh when the spatial adaptivity is needed.

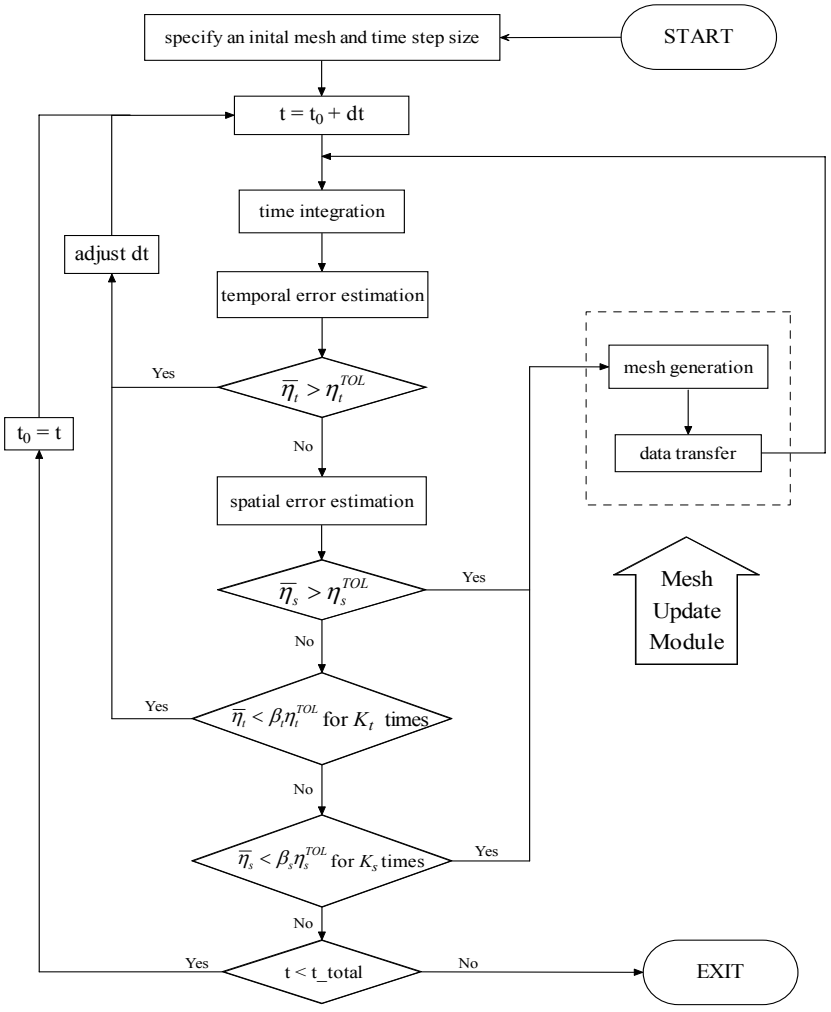


Figure 6.2: Flow chart of the s-adaptive procedure for linear transient analysis

6.2 Numerical Examples

Two simple 2-D example problems are presented to demonstrate the performance characteristics of the structured, multilevel, s-adaptive procedure in linear transient analysis. The 2-D, linear 4-node quadrilateral element is used for both examples.

6.2.1 Propagation of an Axial Stress Wave in a Rectangular Bar

This common example problem involves the propagation and reflection of a compressive normal stress wave in a long, narrow rectangular bar. Figure 6.3 shows the geometry, material properties, constraints and applied loading for the problem. The externally applied normal compression ***p***(t) is uniform in space, but varies with respect to time in the form of a half-sine-wave pulse.

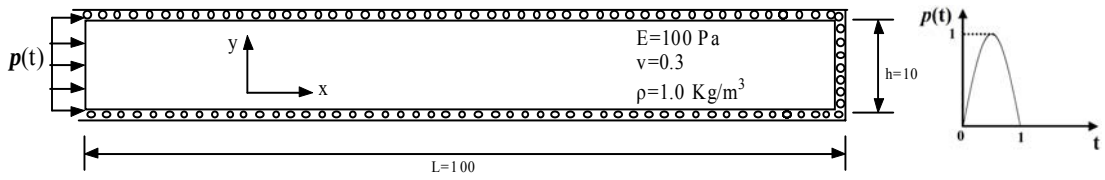


Figure 6.3: A rectangular bar under a half-sine-wave compressive impulse

While this problem admits a 1-D solution, it will be solved as a 2-D, undamped, plane stress problem for illustration purposes. The s-adaptive solution is initiated from a uniform global mesh that contains 20 elements in the x direction and 2 elements in the y direction. Adaptivity of the structured mesh superposition solution is controlled by the

following parameters: temporal error tolerance $\eta_t^{TOL} = 0.1\%$, $\beta_t = 0.6$, $\theta_t = 0.9$, $K_t = 8$, spatial error tolerance $\eta_s^{TOL} = 3\%$, $\beta_s = 0.6$, $\phi = 0.6$, $\theta_s = 0.9$, and $K_s = 8$. The overlay mesh parameters are chosen as $d_h = 4$ and $d_v = 1$. The adaptive solution is computed for times $0 \leq t \leq 15$ sec., which is sufficient for the stress wave to reflect off the fixed end and travel half-way back to its starting point.

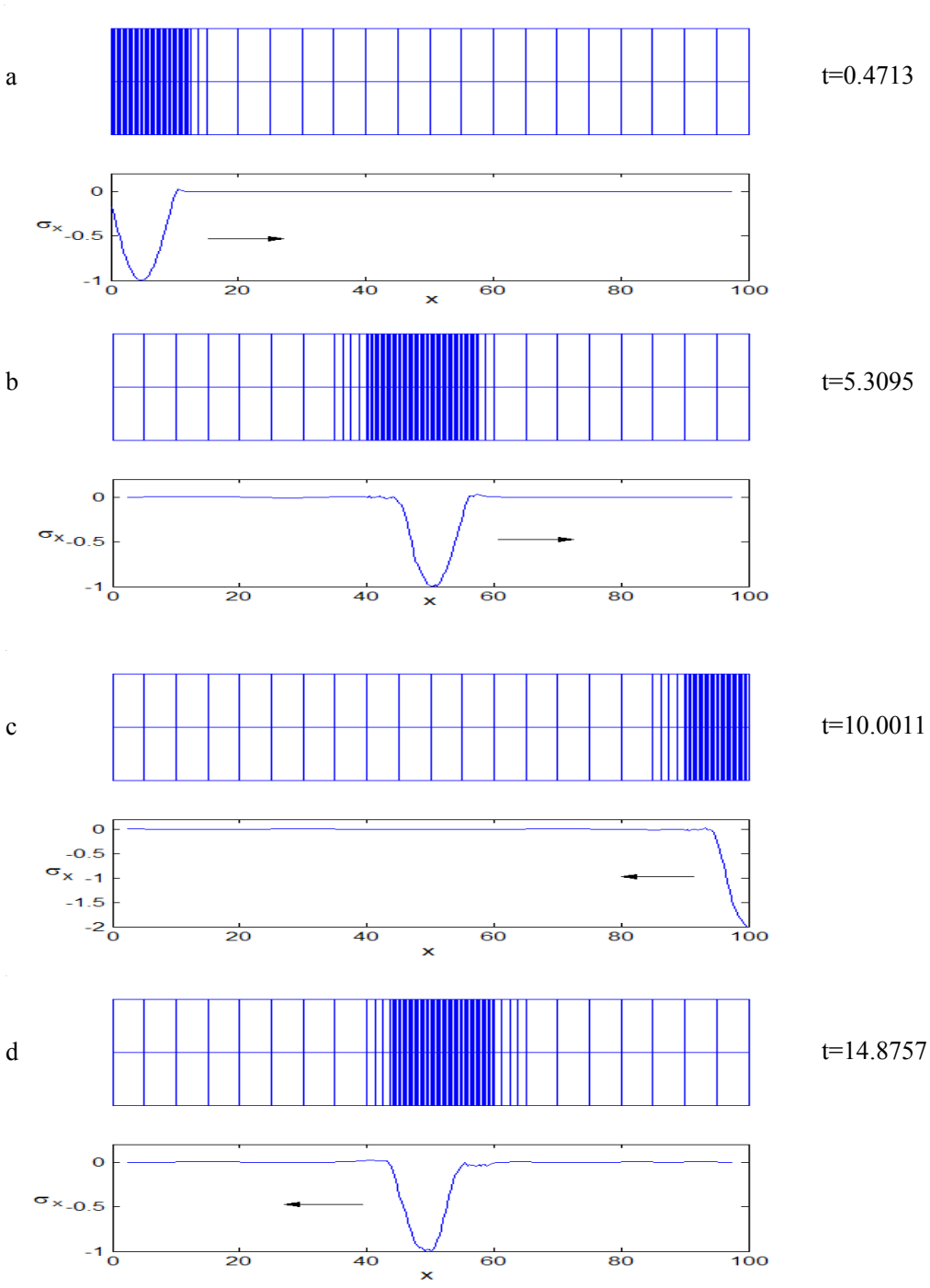


Figure 6.4: Series of composite meshes and computed axial stress distributions produced by the structured, s-adaptive solution at various points in time

Figure 6.4 shows a series of composite meshes and associated axial stress distributions that were produced at various points in time during the s-adaptive analysis. As seen in Figure 6.4, the overlay mesh is repeatedly updated to provide a composite mesh with high refinement in the immediate vicinity of the traveling stress wave, while maintaining a coarse mesh elsewhere. In order to maintain the user-specified spatial error tolerance, two levels of overlay meshes were needed, resulting in the smallest elements having a length of L/320 in the direction of wave propagation.

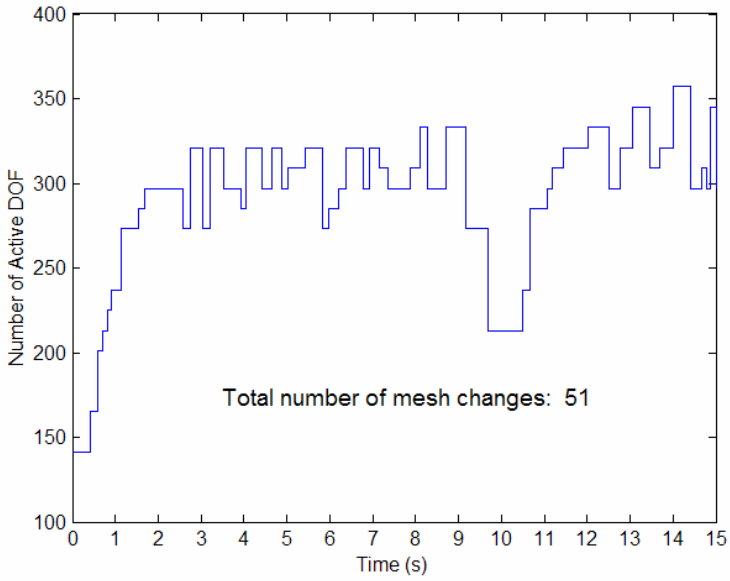


Figure 6.5: Time history of the number of active DOF used in the s-adaptive analysis

Figure 6.5 shows the number of active DOF used in the composite mesh as a function of time. Note that after the stress wave becomes fully formed within the bar (at approximately t = 1 sec.), the number of active DOF remains relatively stable, except

during the time period when the stress wave reflects off the far end of the bar (9 sec. < t < 11 sec.). Note also that in response to the estimated spatial error, the composite mesh was updated 51 times during the course of the analysis; however, each update involves only minor changes to the overlay mesh and can be performed very quickly. The estimated relative spatial error is presented in Figure 6.6.

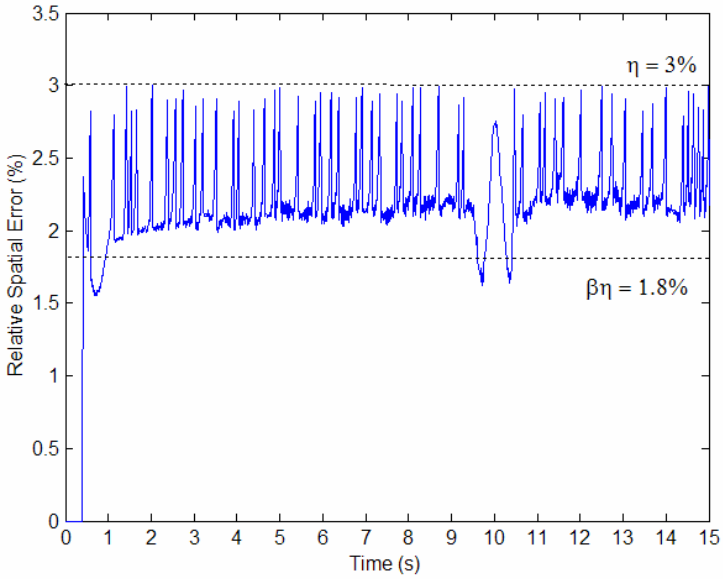


Figure 6.6: Time history of relative spatial error achieved in the adaptive analysis

The estimated temporal error is used to control the time step size over the course of the adaptive analysis. Figure 6.7 shows the time history of the time step size, while Figure 6.8 shows the estimated temporal error.

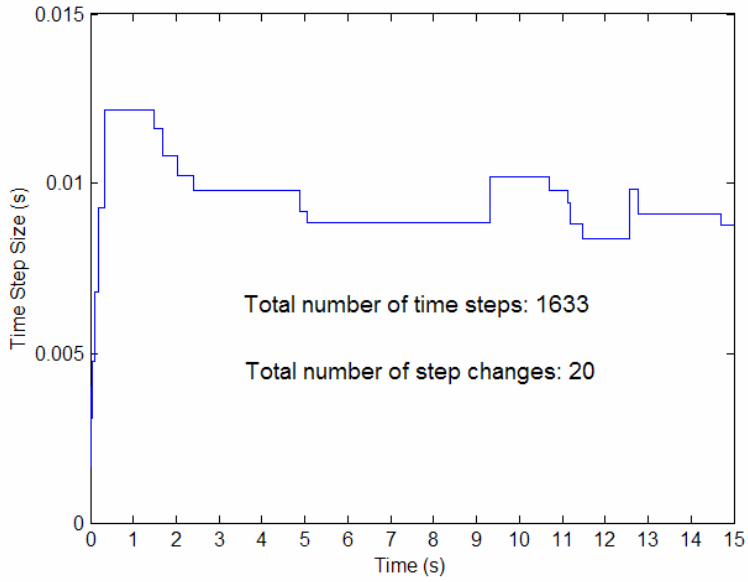


Figure 6.7: Time history of the time step size used in the adaptive analysis

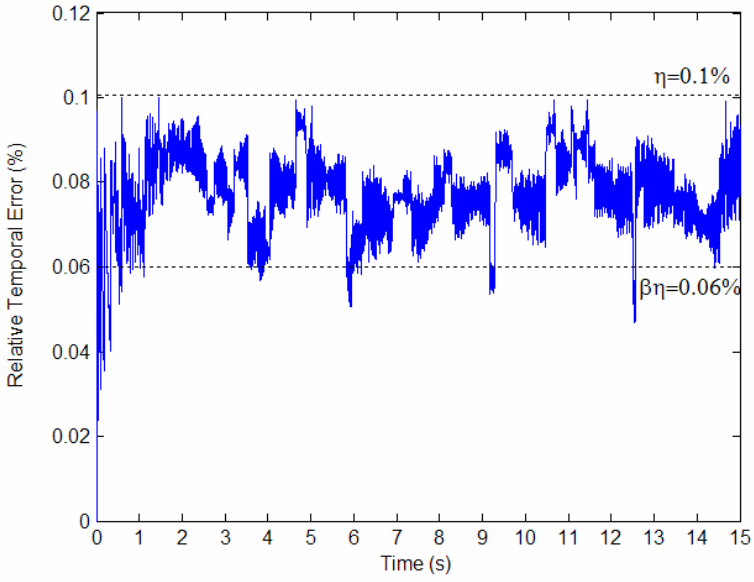


Figure 6.8: Time history of relative temporal error achieved in the adaptive analysis

Compared to similar results obtained by Wiberg et al. [45], where spatial adaptivity was performed by reformulating the entire mesh, the present method produces a more consistent stress wave form (i.e. amplitude and shape) and less spurious oscillation.

This improvement can be attributed to the simple process used by the structured mesh superposition method in transferring the current solution from the current composite mesh to the next composite mesh. Traditional methods that use a reformulation of the entire mesh necessarily require a large portion of the solution to be *interpolated* onto the new mesh. In a transient analysis, this interpolation consistently and effectively introduces local errors in the initial conditions that are used to integrate the solution from the current time t to time $t+\Delta t$, thus introducing additional spurious oscillation into the solution. However, in the structured mesh superposition method, there is no interpolation required for regions where the mesh density is being increased. Thus the only error that occurs during solution transfer is the unavoidable error caused by discarded overlay DOF in regions that undergo coarsening.

For comparison, the same problem is solved with a uniform refined mesh of 320 elements in the x direction and 2 elements in the y direction (1281 active DOF) with a time step size dt=0.01 sec. The uniform element size is comparable to the most refined overlay mesh level used previously in the s-adaptive solution. Figures 6.9 and 6.10 show the location, size and shape of the compressive normal stress wave predicted by both methods at the discrete times t=1, 4, 8, 10 and 15 sec. Despite the fact that the s-adaptive solution utilizes approximately one-fourth the number of DOF used by the uniform refined solution, the two solutions show excellent agreement.

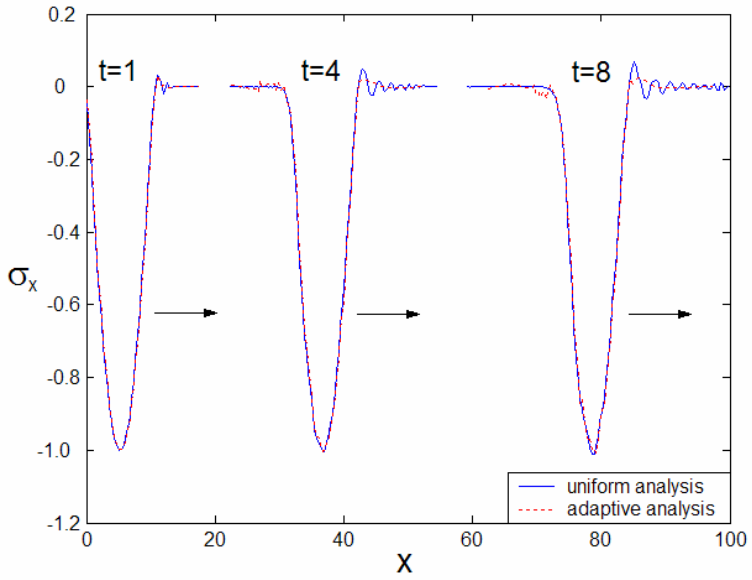


Figure 6.9: Comparison of the compressive normal stress wave predicted at discrete times t=1,4,8 sec. by the s-adaptive solution and the uniform refined solution

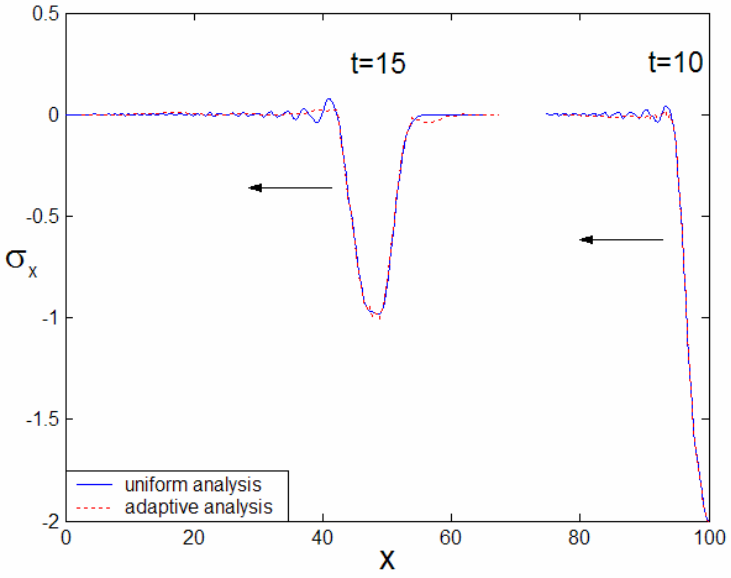


Figure 6.10: Comparison of the compressive normal stress wave predicted at discrete times t=10, 15 sec. by the s-adaptive solution and the uniform refined solution

A close examination of the spurious oscillations at the leading and trailing edges of the stress wave reveals some qualitative differences between the two solution methods. Figure 6.11 shows a zoom-in view of the compressive stress wave at time t=6 seconds. The solution obtained with the uniform refined mesh exhibits significant high frequency oscillation at the leading edge of the wave. This behavior is typical in conventional displacement-based finite element models that are implicitly integrated. The frequency of the spurious oscillation is dependent of the size of the elements ahead of the stress wave. In contrast, the s-adaptive method does not require the use of highly refined overlay meshes in the region ahead of the leading edge of the stress wave; therefore, this region remains coarsely meshed with relatively large elements, causing the spurious oscillation to be lower in both frequency and amplitude.

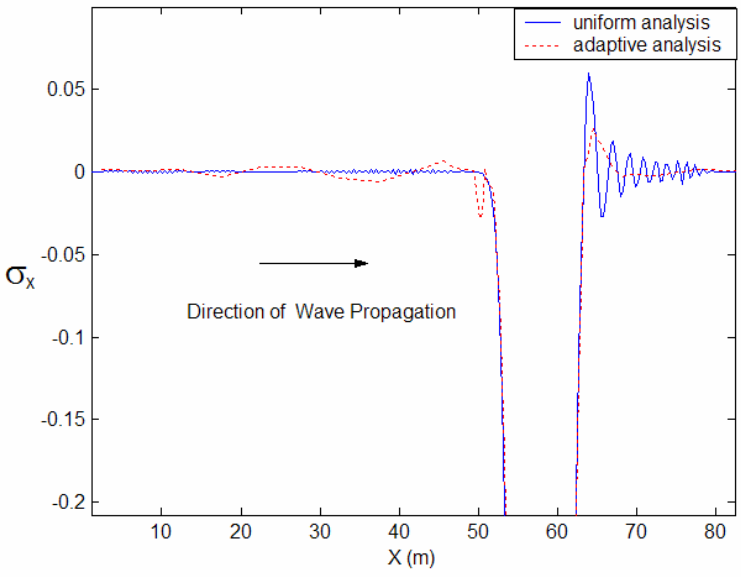


Figure 6.11: Close-up view of the compressive stress wave showing areas of spurious oscillation

At the trailing edge of the stress wave, the situation is reversed, i.e. the s-adaptive method exhibits more spurious oscillation than the uniform refined solution. This is mainly caused by the process of coarsening the mesh in the subregion behind the trailing edge. During mesh coarsening, some of the incremental overlay DOFs are simply discarded, resulting in a local, momentary imbalance which introduces additional spurious oscillation into the transient solution. Fortunately, this interpolation error is small since the mesh is only coarsened when the estimated spatial error is very small.

6.2.2 Damped, Free Vibration of a Thick Clamped/Clamped Beam

This example is a typical structural dynamics problem and involves a clamped/clamped beam that that is set into damped free vibration by the instantaneous removal of a uniform distributed transverse load. Figure 6.12 shows the geometry, boundary and initial conditions, material properties, Rayleigh damping coefficients and external loading. In contrast to the previous wave propagation problem that exhibited very pronounced localization, the adaptivity required in the present structural dynamics problem is more global in nature. A uniform global mesh is chosen with 12 elements in the x direction and 4 elements in the y direction to initiate the s-adaptive analysis. Time integration parameters are selected as α=0, β=0.3025 and γ=0.6. Relative error tolerances and corresponding parameters are specified as: $\eta_t^{TOL} = 0.1\%$, β_t = 0.5, θ_t = 0.8, K_t =10 and $\eta_s^{TOL} = 10\%$, β_s = 0.6, ϕ = 0.6, θ_s = 0.9, K_s =10 respectively. The overlay mesh

parameters are chosen as $d_h=d_v=2$. Computation is conducted over a time interval $0\leq t\leq 17$ sec., which is sufficient for two complete vibration cycles.

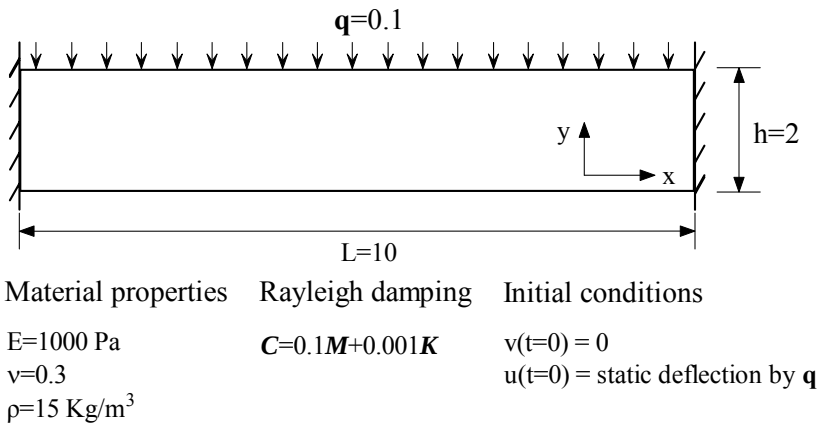


Figure 6.12: A damped, free vibration of a clamped/clamped beam after instantaneous removal of a static, uniform, distributed transverse load

Upon removal of the static loading, the beam begins vibrating in its fundamental mode, which is characterized by high levels of curvature at the two clamped boundaries and at the center of the span. The composite meshes that are generated during the analysis necessarily exhibit higher levels of refinement in these regions. Furthermore, the overall density of the composite mesh is typically highest when the beam is at maximum deflection and lowest when the beam passes through its undeformed configuration. In order to maintain the user-specified spatial error tolerance, the s-adaptive method used up to two levels of overlay mesh, resulting in the smallest elements having a dimension of $L/48 \times h/16$.

Figure 6.13 shows the series of the composite meshes that were used during various time intervals over the course of the analysis. Figure 6.13(a) shows the transverse deflection of the beam's midpoint on the top surface and the number of active DOF as functions of time. The composite mesh was updated 21 times over a period of two complete vibration cycles. Notice that there is a very pronounced correlation between transverse deflection and composite mesh density. Figure 6.13(b) shows the composite mesh that is used during the initial departure from the statically deformed configuration. This is the highest level of composite mesh density that was required in the analysis and corresponds to the highest level of deflection achieved in the analysis. Notice that the two regions of lower mesh density correspond to the locations of inflection points where the beam curvature is low. Figures 6.13(c) and 6.13(d) are relatively coarse composite meshes that were adequate as the beam approached its undeformed configuration during its vibration. Figure 6.13(e) shows a moderate level of composite mesh density that was used during an interval of intermediate deflection level. Figures 6.13(a), 6.13(f) and 6.13(h) show the composite meshes used at peak deflections during the damped oscillation where we see that the mesh density exhibits a very gradual decrease that is commensurate with the decrease in deflection amplitude caused by the damping. The spatial adaptivity is driven by the estimated spatial error shown in Figure 6.14.

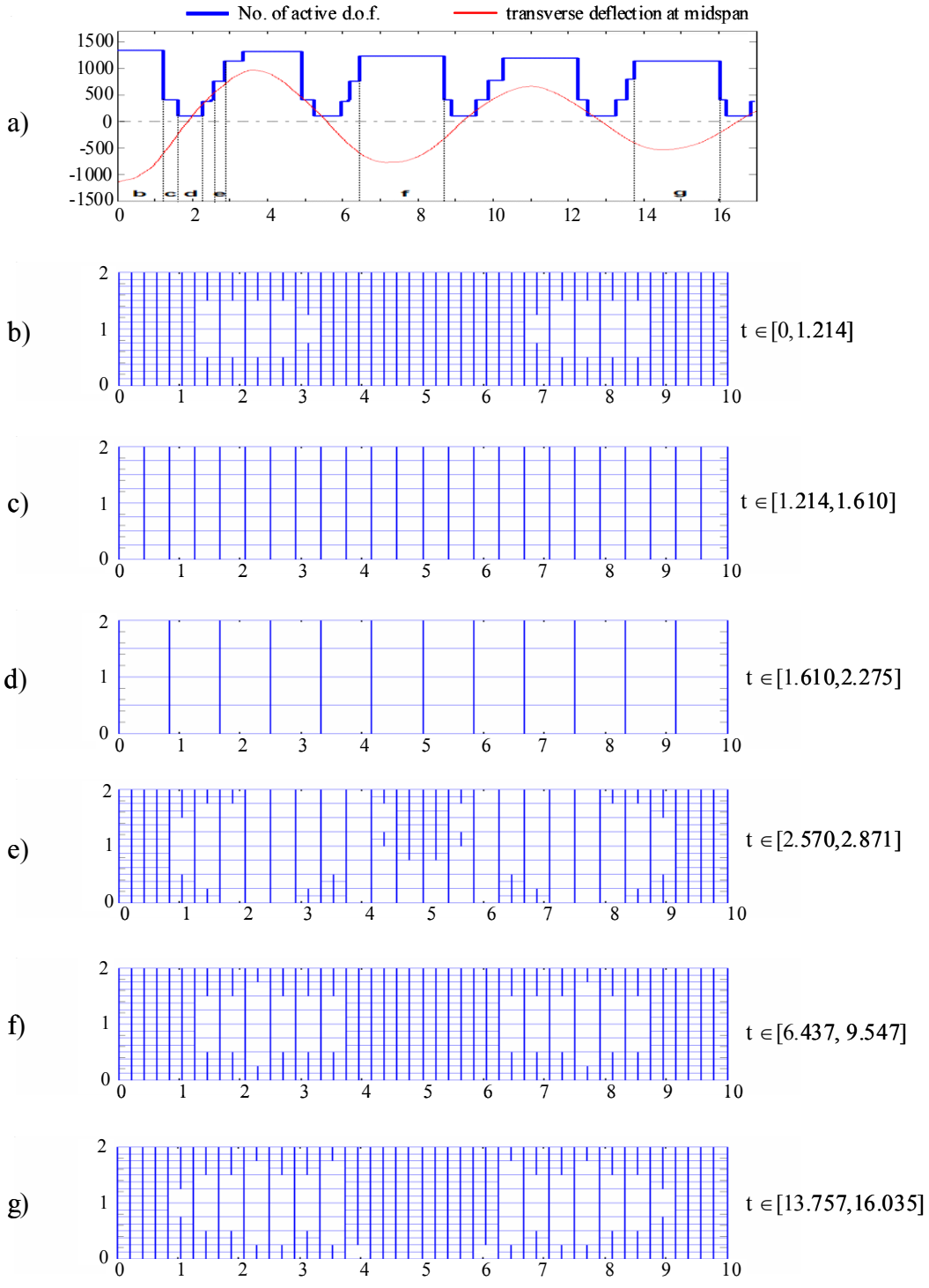


Figure 6.13：Series of composite meshes used in the s-adaptive analysis of a clamped/clamped beam under damped free vibration

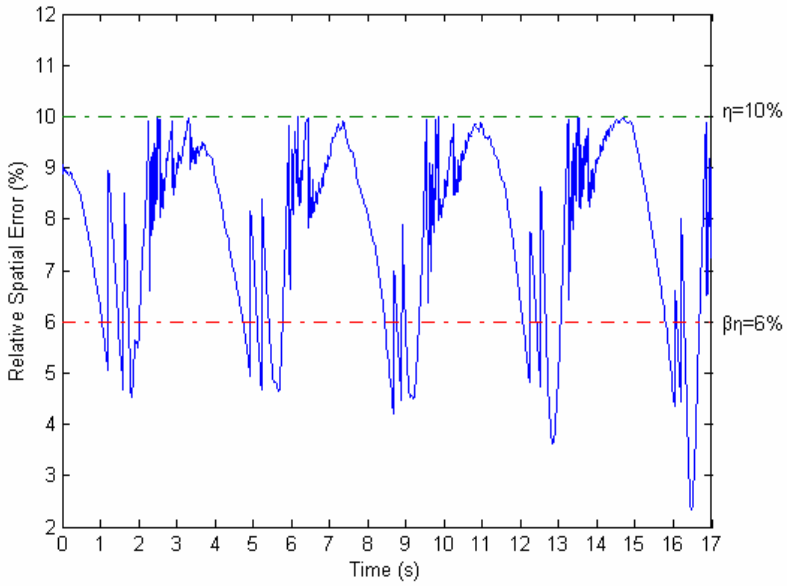


Figure 6.14: Time history of relative spatial error achieved in the adaptive analysis

Figures 6.15 and 6.16 show the required time step size and relative temporal error as functions of time.

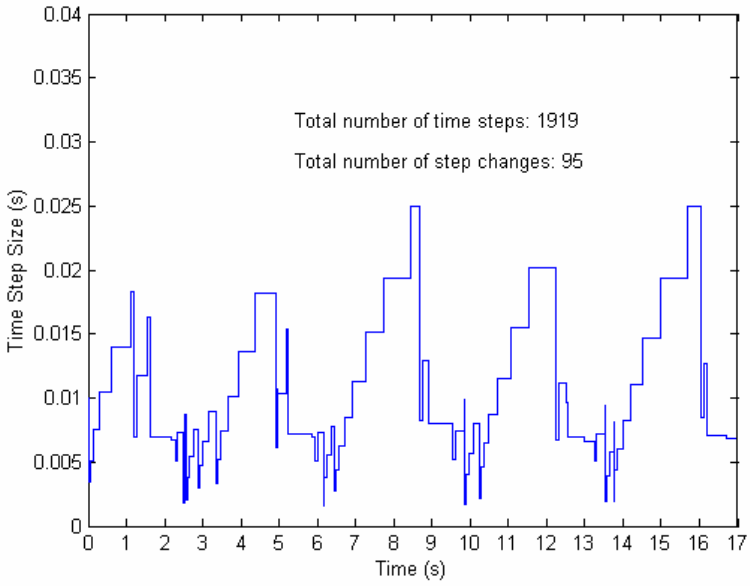


Figure 6.15: Time history of the time step size used in the s-adaptive analysis

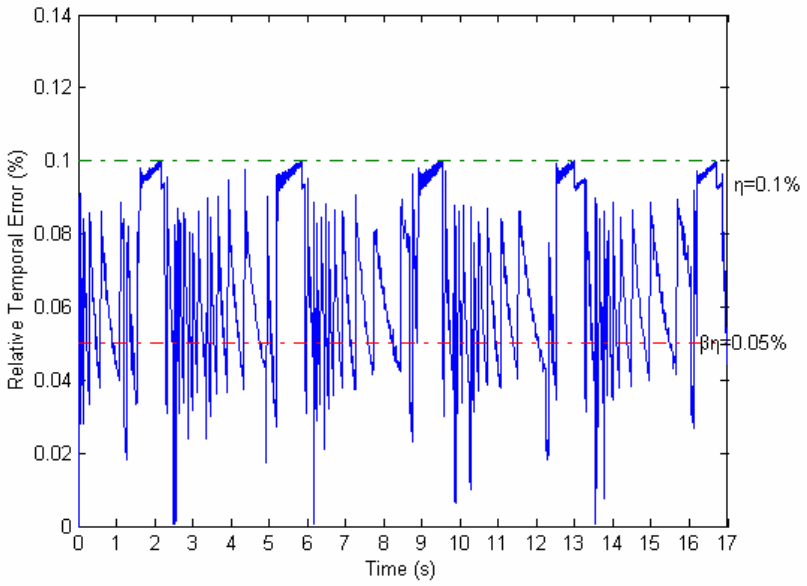


Figure 6.16: Time history of relative temporal error in the s-adaptive analysis

In comparing Figure 6.13(a) and 6.15, we see that the required time step size begins to increase whenever the composite mesh density remains constant for any extended period of time, while the required time step size decreases rapidly when abrupt changes are made to the composite mesh density.

For comparison, the problem is also solved using a uniform refined mesh of 48 elements in the x direction and 14 elements in the y direction (1410 active DOF) with a time step size dt=0.01 seconds. In this case, the uniform element size is comparable to the elements used in the most refined overlay mesh in the s-adaptive analysis. Figure 6.17 shows that the transverse deflections predicted by both solutions show close agreement

despite the fact that the s-adaptive solution uses only a fraction of the number of degrees of freedom used by the uniform refined solution.

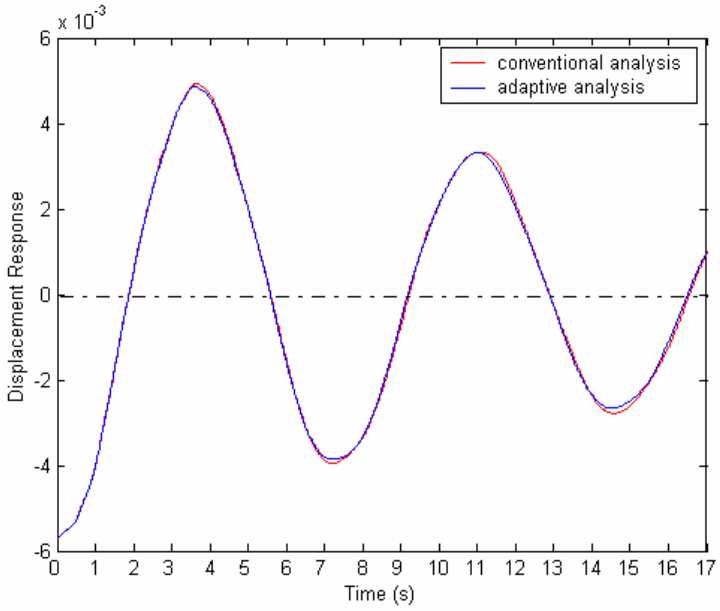


Figure 6.17: Time history of the vertical displacement at top center of the beam

In this problem, the presence of the Rayleigh damping dissipates the total energy of the beam as in shown Figure 6.18, causing each successive deflection amplitude to decrease dramatically. In this study, the estimates for the relative error were computed based on the instantaneous total energy norm. This choice permits accuracy to be maintained even after the amplitude of vibration has significantly diminished. If the maximum of total energy norm during the calculation is used in evaluating the estimated relative error (e.g. [45]), obviously, less accurate solution will be accepted which is confirmed by the comparison as shown in Figure 6.19.

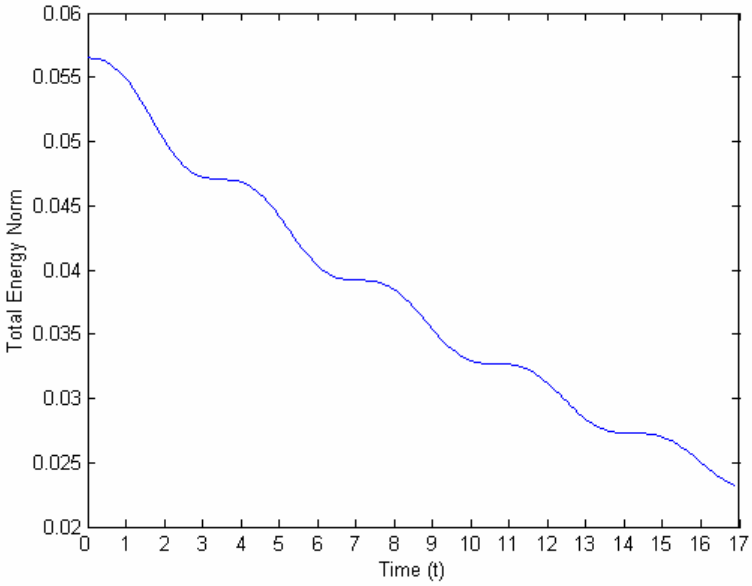


Figure 6.18: Time history of the total energy norm of the beam

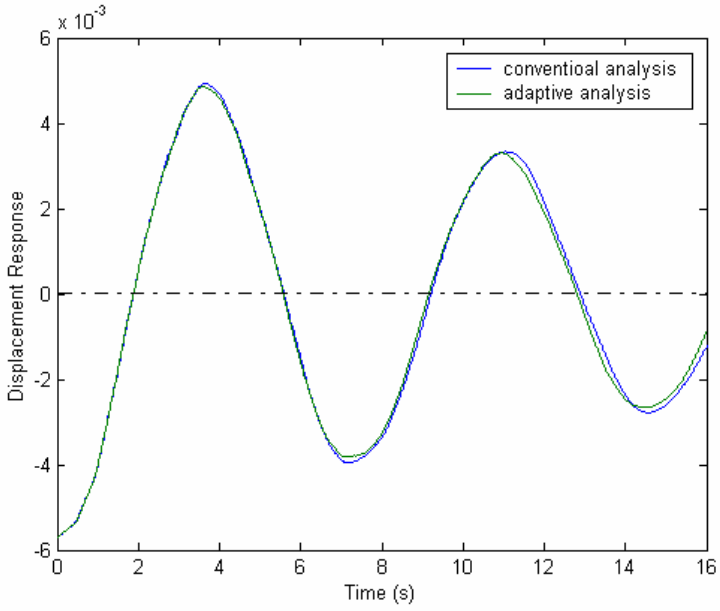


Figure 6.19: Time history of the vertical displacement at top of center of the beam

Chapter 7: STRUCTURED S-ADAPTIVITY IN NONLINEAR ELASTODYNAMICS CAUSED BY PROGRESSIVE DAMAGE

The encouraging performances of the structured mesh superposition method in linear dynamic analysis in terms of high efficiency, robustness and accuracy inspire its application in more computationally demanding nonlinear dynamic analysis. In this study, the problems under consideration are limited to nonlinear material behavior in the form of elastic progressive damage, but under the assumption of small displacement, which implies small deformations and small rotations. The enhancement of the s-adaptive method to encompass material nonlinear dynamic problems is a challenging task that involves several difficulties as briefly outlined in Chapter 1. In this chapter, some background and introduction on progressive damage and the implicit nonlinear finite element method are first presented in section 7.1; the proposed methodology to overcome the difficulties is presented in section 7.2, followed by five numerical examples in section 7.3.

7.1 Background and Introduction

7.1.1 Progressive Damage

Progressive Damage is the source of the materially nonlinearity in this study. On the microscopic scale, damage is characterized by molecular bonds that have been broken and are no longer available to support loads. Typically these broken bonds are not uniformly distributed in the material, tending instead to nucleate into microcracks and microvoids. As damage accumulates, the molecular bond density of the material decreases, resulting in a reduction of stiffness and consequently a nonlinear (softening) stress/strain relationship. In essence, progressive damage is analogous to plasticity in terms of the analytical approaches required to model these physical processes. Despite the similarities between progressive damage and plasticity, unlike material involved in plasticity, damaged material does not present permanent deformation.

A phenomenological approach known as continuum damage mechanics (CDM) [79] has emerged for the development of damage-dependent constitutive relations. The implementation of the capability to model microscopic damage via CDM requires the following four key components, which has been summarized by Reddy and Robbins [80]. Obviously, in each aspect of the progressive damage, there have been various assumptions or schemes proposed in the literature. For better understanding of the

proposed methodology and the subsequent numerical examples in this chapter, these important components of CDM are briefly cited from [80] and the focus is on the specific choices made or schemes adopted in this study.

Damage Variable—An appropriate damage tensor must be chosen to characterize the state of microscopic damage by defining the density and orientation of microcracks and microvoids. Various forms of the damage tensor, such as scalars, vectors, 2^{nd} order tensors, and 4^{th} order tensors, have been proposed in the literature [81, 82]. In this study, a symmetric 2^{nd} order damage tensor **D** is adopted, which implies that the microscopic damage is orthotropic. When the damage tensor is expressed in its principal coordinate system, its eigenvalues (denoted as D_1, D_2 in 2-D problems) have a simple physical interpretation. The i^{th} eigenvalue D_i represents the fractional reduction in load carrying area on planes that are perpendicular to the i^{th} principal direction. The eigenvalues of the damage tensor must be in the range $0 \leq D_i \leq 1$ where $D_i = 0$ corresponds to a complete lack of microcracks, while $D_i = 1$ corresponds to a complete separation of the material. In this study, the damage in a 2-D linear quadrilateral element is assumed to be uniformly distributed within the element and is represented by the damage eigenvalues evaluated at the single reduced integration point within the element.

Stiffness Reduction Scheme—Given the original undamaged material properties, the stiffness of the material must be reduced in a way that is consistent with the current level

and type of microscopic damage as defined by the damage tensor. This stiffness reduction process is based on various equivalence principles, for example, the principle of equivalent elastic strain energy density adopted in this study. For an orthotropical material characterized by the 2-D elasticity tensor $\mathbf{C}$ in (7.1a), the damaged elasticity tensor $\bar{\mathbf{C}}$ is given by (7.1b) as

$$[\mathbf{C}] = \begin{bmatrix} C_{11} & C_{12} & 0 \\ C_{21} & C_{22} & 0 \\ 0 & 0 & C_{33} \end{bmatrix}, \quad [\bar{\mathbf{C}}] = \begin{bmatrix} \bar{C}_{11} & \bar{C}_{12} & 0 \\ \bar{C}_{21} & \bar{C}_{22} & 0 \\ 0 & 0 & \bar{C}_{33} \end{bmatrix} \tag{7.1a,b}$$

where

$$\bar{C}_{\alpha\beta} = C_{\alpha\beta}\left(1 - D_{\alpha}\right)\left(1 - D_{\beta}\right) \qquad (\alpha, \beta = 1, 2) \tag{7.2a}$$

$$\bar{C}_{33} = C_{33}\left(1 - D_1\right)\left(1 - D_2\right) \tag{7.2b}$$

Damage Surface—Like the yield function used to distinguish elastic deformation from plastic deformation in plasticity theory, the damage surface, i.e. a multidimensional convex surface g($\mathbf{Y}$, γ) = 0 that is in the space of state variables, serves to distinguish non-damaging behavior from damage-inducing behavior. The damage surface is often assumed to be a quadratic function of the energy release rate tensor $\mathbf{Y}$, which is a function of the undamaged material properties, the strain and the damage tensor. A damage hardening variable γ is also needed to define the evolving damage surface. In this study, the damage surface for 2-D problems is expressed in the principal material coordinate system as

$$g(\mathbf{Y},\gamma)=\frac{1}{2}\left[J_{11}(Y_1)^2+J_{22}(Y_2)^2\right]-\gamma \tag{7.3}$$

where the material constants J_{11} and J_{22} defines the damage tolerance of the material and control the shape of the damage surface. In the present study, a linear damage hardening function $\gamma=(B_0+B)=(B_0+c_1\beta)$ is used to define the current damage threshold and thus controls the size of the damage surface, where the material constant B_0 defines the damage threshold of the un-damaged material, and the term $c_1\beta$ provides the increase in the damage threshold, (i.e. damage hardening) that occurs as the damage tensor **D** evolves. The overall damage parameter β is a scalar variable used to define the overall state of damage for the purpose of evaluating damage hardening.

Damage Evolution—Analogous to the elasto-plasticity analysis where the plastic deformation is occurring on the evolving yield surface, the damage progress is allowed to occur only on the evolving damage surface. When a point in the material exhibits a state which results in the condition $g(\mathbf{Y}, \gamma)<0$, then no further damage occurs at that point. However, when the deformations at a point in the material cause the state of the material to yield the condition $g(\mathbf{Y}, \gamma)>0$, it results in a simultaneous increase in the damage tensor **D** and the overall damage parameter β in such a way as to maintain the condition $g(\mathbf{Y},\gamma)=0$ throughout the damaging process. The damage evolution equations express increments in the damage tensor **D** and increments in the overall damage parameter β in terms of the current state variables $(\boldsymbol{\varepsilon},\mathbf{D},\beta)$ and the current imposed strain increments $d\boldsymbol{\varepsilon}$. The evolution equations are developed with the framework of irreversible

thermodynamics, specifically making use of the principle of maximum dissipation. For the details of the derivation, the reader is referred to Reddy and Robbins [80].

7.1.2 Implicit Nonlinear Finite Element Method

A brief introduction of the formulation of implicit nonlinear finite element method [63] is described in this section. The derivation of the implicit nonlinear finite element method using HHT-α variant of Newmark time integration method is provided in Appendix A.

After spatial discretization using finite elements, the equations of motion of a continuum can be transformed into a coupled system of second order nonlinear ordinary differential equations as

$$\mathbf{M}\ddot{\mathbf{u}}(t)+\mathbf{C}\dot{\mathbf{u}}(t)+\mathbf{f}^{\text{int}}(\mathbf{u})=\mathbf{f}^{\text{ext}} \tag{7.4}$$

where **M** is the mass matrix, **C** is the damping matrix, $\mathbf{f}^{\text{int}}$ is the internal nodal force vector, $\mathbf{f}^{\text{ext}}$ is the external nodal force vector. The internal force vector $\mathbf{f}^{\text{int}}$ is assembled from the element internal force vector $\mathbf{f}_e$ which is given by

$$\mathbf{f}_e = \int_{\Omega_e} \mathbf{B}^T \boldsymbol{\sigma} d\Omega \tag{7.5}$$

where **B** is the strain shape function matrix [5] and Ω_e is the element domain.

The implicit finite element method assumes that state variables satisfy the equation (7.4) at any time. In the time step $[t_n, t_{n+1}]$, it assumes the state variables are known at time t_n in order to solve for the solution of the state variables at time t_{n+1}. Therefore, the solution procedure aims to find the state variables that satisfy

$$\mathbf{M}\ddot{\mathbf{u}}_{n+1} + \mathbf{C}\dot{\mathbf{u}}_{n+1} + \mathbf{f}^{\text{int}}(\mathbf{u}_{n+1}) = \mathbf{f}^{\text{ext}}_{n+1} \tag{7.6}$$

where $\mathbf{u}_{n+1}$, $\dot{\mathbf{u}}_{n+1}$, $\ddot{\mathbf{u}}_{n+1}$ and $\mathbf{f}^{\text{ext}}_{n+1}$ are the nodal displacements, velocities, accelerations and external loads vectors evaluated at time t_{n+1}, respectively. Generally, linearization and iteration are needed to solve the nonlinear equation (7.6). First, the equation is linearized about the configuration at time t_n. We express $\mathbf{u}_{n+1}$ as follows

$$\mathbf{u}_{n+1} = \mathbf{u}_n + \Delta\mathbf{u}^0 \tag{7.7}$$

where $\Delta\mathbf{u}^0$ is the displacement increment in the time step $[t_n, t_{n+1}]$.

Substitution of equation (7.7) into equation (7.6) and yield the following linearized discrete equation.

$$\mathbf{M}\ddot{\mathbf{u}}_{n+1} + \mathbf{C}\dot{\mathbf{u}}_{n+1} + \mathbf{K}(\mathbf{u}_n)\Delta\mathbf{u}^0 = \mathbf{f}^{\text{ext}}_{n+1} - \mathbf{f}^{\text{int}}(\mathbf{u}_n) \tag{7.8}$$

where $\mathbf{K}(\mathbf{u}_n)$ is the tangent stiffness matrix evaluated at the configuration at time t_n.

Equation (7.8) may be further transformed using implicit time integration scheme such as Newmark method or HHT-α method, where the unknown quantities $\dot{\mathbf{u}}_{n+1}$ and $\ddot{\mathbf{u}}_{n+1}$ are

expressed in terms of $\mathbf{u}_{n+1}$ and the known quantities $\mathbf{u}_n$, $\dot{\mathbf{u}}_n$ and $\ddot{\mathbf{u}}_n$, such as in equation (2.4b,c). Ultimately, the subsequent use of equation (7.7) in the resulting equations from time integration yields a system of linear equations as

$$\mathbf{K}^{\text{eff}}(\mathbf{u}_n)\Delta\mathbf{u}^0 = \mathbf{f}_{n+1}^{\text{ext}} - \mathbf{f}^{\text{eff}}(\mathbf{u}_n) = \mathbf{f}^0 \tag{7.9}$$

where $\mathbf{K}^{\text{eff}}$ and $\mathbf{f}^{\text{eff}}$ are the effective stiffness matrix and the effective internal force vector respectively, both of which are functions of displacement. The residual force vector $\mathbf{f}^0$ is defined to be the difference between the external force and internal force.

After solving for $\Delta\mathbf{u}^0$, the displacement at time t_{n+1} is updated using

$$\mathbf{u}_{n+1}^1 = \mathbf{u}_n + \Delta\mathbf{u}^0 \tag{7.10}$$

In general, the residual force vector $\mathbf{f}^1 = \mathbf{f}_{n+1}^{\text{ext}} - \mathbf{f}^{\text{eff}}(\mathbf{u}_{n+1}^1)$ is not a zero vector in nonlinear problems. This means that the equation (7.8) is not satisfied by the computed $\mathbf{u}_{n+1}^1$ and equilibrium iterations are performed by solving equation (7.11) and updating the displacement by equation (7.12) to obtain the solution which satisfies equation (7.8).

$$\mathbf{K}^{\text{eff}}(\mathbf{u}_{n+1}^i)\Delta\mathbf{u}^i = \mathbf{f}_{n+1}^{\text{ext}} - \mathbf{f}^{\text{eff}}(\mathbf{u}_{n+1}^i) \qquad (\text{i}=0,1,2,\cdots) \tag{7.11}$$

$$\mathbf{u}_{n+1}^{i+1} = \mathbf{u}_{n+1}^i + \Delta\mathbf{u}^i \quad (\text{i}=0,1,2,\cdots) \tag{7.12}$$

where *i* denotes the number of iterations and $\mathbf{u}_{n+1}^0 = \mathbf{u}_n$. The iterations of equation (7.11) and equation (7.12) are performed until convergence has been reached, which stands for the successful obtaining of the solution of the state variables at time t_{n+1}.

7.1.3 CDM Module

In order to simulate the dynamic problems with material nonlinearity caused by progressive elastic damage, a *CDM Module* developed by Robbins [80] is utilized in the implicit nonlinear finite element method. Figure 7.1 is a brief flowchart of the nonlinear finite element procedure that simulates quasi-static elastic progressive damage using CDM Module.

In Figure 7.1, the *CDM Module* is a set of FORTRAN subroutines that encapsulates the relevant Continuum Damage Mechanics (CDM) equations. The *CDM Module* greatly facilitates the implementation of CDM into the existing finite element codes. In the CDM Module, the major input is the strain increment and the direct outputs are the damage tensor increment, overall damage hardening parameter increment and the damaged stiffness matrix caused by the imposed strain increment.

Step 1. Increment the externally imposed load by an amount $\{d\mathbf{f}\}_{ext}$ to yield a cumulative imposed load vector of $\{\mathbf{f}\}_{ext} = \{\mathbf{f}\}_{ext} + \{d\mathbf{f}\}_{ext}$

Step 2. Based on the current damaged stiffness matrix $[\widetilde{\mathbf{C}}]$, and current level of cumulative strain $\{\boldsymbol{\varepsilon}\}$, compute the constitutive stress $\{\boldsymbol{\sigma}\}_{con} = [\widetilde{\mathbf{C}}]\{\boldsymbol{\varepsilon}\}$

Step 3. Based on the constitutive stress $\{\boldsymbol{\sigma}\}_{con}$, compute the element internal force vector and then assemble them to global internal force vector $\{\mathbf{f}\}_{int}$

Step 4. Compute the global residual force vector $\{\mathbf{f}\}_{res} = \{\mathbf{f}\}_{ext} - \{\mathbf{f}\}_{int}$

Step 5. *Elastic predictor step*. Solve the linearized incremental equation $[\mathbf{K}]\{d\mathbf{u}\} = \{\mathbf{f}\}_{res}$ for the global displacement increment.

Step 6. Based on the global displacement increment $\{d\mathbf{u}\}$, compute the corresponding strain increment $\{d\boldsymbol{\varepsilon}\}$ at each integration point.

Step 7. *Damage corrector step*. Call the CDM module to perform the following functions at each integration point:

7a) Use the strain increment $\{d\boldsymbol{\varepsilon}\}$ to solve the damage evolution equations for the damage increment $\{d\mathbf{D}\}$ and damage hardening parameter incremenent $d\beta$.

7b) Update $\{\mathbf{D}\} = \{\mathbf{D}\} + \{d\mathbf{D}\}$; $\beta = \beta + d\beta$; $\boldsymbol{\varepsilon} = \boldsymbol{\varepsilon} + d\boldsymbol{\varepsilon}$

7c) Update the damaged stiffness matrix $[\widetilde{\mathbf{C}}]$.

Step 8. Check for convergence of the residual force vector $\{\mathbf{f}\}_{res}$, the cumulative strain $\{\boldsymbol{\varepsilon}\}$, and the cumulative damage $\{\mathbf{D}\}$. If convergence is achieved, then go to **Step 1** and apply the next imposed load increment. If convergence is not achieved, then continue repeating **Steps 2** through **Step 8** until convergence is achieved.

Figure 7.1: Flowchart of the nonlinear finite element procedure that simulates quasi-static elastic progressive damage using CDM Module

7.2 Proposed Methodology

The goal of this study is to develop an efficient adaptive finite element procedure for dynamic problems with material nonlinearity caused by progressive damage by applying

structured mesh superposition to provide the necessary spatial adpativity. In developing such an adaptive methodology for this particular class of problems, several difficult issues must be resolved. These issues include the development of a suitable error estimation method, the development of a theoretically sound procedure for transferring data from one mesh to the next, and the development of efficient methods for constructing the tangent system of equations for the composite mesh. This section is thus devoted to a discussion of the proposed methods for resolving these difficulties.

7.2.1 Error Estimation

For nonlinear problems, different recovery/postprocessed types of spatial error estimators are available in the literature depending upon the characteristics of the problem under consideration and the objective of the adaptive FE analysis. In the present study of s-adaptive analysis of dynamic problems with material nonlinearity caused by progressive damage, the goal is not to develop a new error estimator for this class of problems, but rather to demonstrate that s-adaptive methodology can be effectively applied to this class of problems. Therefore, for dynamic problems with material nonlinearity, the same basic quantity will be used for error estimation as in the linear problems, namely, the total energy norm of stress error.

$$\left\|\mathbf{e}_{\mathrm{s}}\right\|_{\mathrm{E}} = \left[\int_{\Omega} (\boldsymbol{\sigma}^{*} - \boldsymbol{\sigma}^{h})^{\mathrm{T}} \mathbf{D}^{-1} (\boldsymbol{\sigma}^{*} - \boldsymbol{\sigma}^{h}) d\Omega\right]^{1/2} \tag{7.13}$$

where **D** is the current (reduced) constitutive matrix from appropriate *stiffness reduction scheme* after damage occurs. The SPR procedure is used to compute $\boldsymbol{\sigma}^*$ which represents the improved (i.e. postprocessed) values of the finite element total stress $\boldsymbol{\sigma}^h$. As pointed out by Wiberg and Li [54], the SPR technique can be valid even though there does not exist superconvergent points for stresses, such as in the context of nonlinear problems. The meaning of SPR has already been generalized to the postprocessing technique of improving FE-solutions by the use of local patches. This choice of this simple and familiar error estimate enables us to focus exclusively on our main goal, the application of mesh superposition technique in adaptive nonlinear transient analysis.

It is worth noting that the present choice of error estimation of total energy norm is perfectly suitable for the nonlinearity caused by progressive damage. As mentioned in section 7.1, damaged material does not present permanent deformation. In an unloading process, damaged material demonstrates a linear elastic behavior though the stiffness of damage material may be smaller than the stiffness of undamaged virgin material. As a result, the total energy norm adopted in this study indeed accurately measures the total energy stored in the damaged material, as long as the constitutive matrix **D** used in equation (7.13) is the current (reduced) constitutive matrix after damage.

The basic adaptive strategy described in Chapter 5 and Chapter 6 is adopted and is based on the following temporal and spatial relative percent errors

$$\bar{\eta}_t = \frac{\left\|\mathbf{e}_t\right\|_E}{\left\|\mathbf{u}^h\right\|_E} \times 100\% \quad \text{and} \quad \bar{\eta}_s = \frac{\left\|\mathbf{e}_s\right\|_E}{\left\|\mathbf{u}^h\right\|_E} \times 100\% \tag{7.14a,b}$$

where $\left\|\mathbf{e}_t\right\|_E$ and $\left\|\mathbf{u}^h\right\|_E$ are defined in Chapter 2. The two relative percent errors are required to be less than their corresponding tolerances at the end of every time step. Otherwise, the temporal adaptivity and/or spatial adaptivity will be performed.

7.2.2 Data Transfer Scheme

It is known that the two kinds of state variables needed to be transferred from the old mesh to the new mesh during mesh changes in the materially nonlinear finite element analysis are: *node-wise* variables and *element-wise* variables. The data transfer scheme presented in Chapter 6, which has been proved successful in the linear problems, is adopted here for the transfer of *node-wise* variables. The *fission and fusion* data transfer scheme by Belytschko [47] and the *mastergrid* scheme proposed in this study are adopted for the transfer of *element-wise* variables.

Although the *fission and fusion* data transfer scheme is able to minimize the material history diffusion due to data transfer [57], it still has room for improvement. Consider a typical case of adaptive analysis. A stress wave propagates forward in a bar, and it gets reflected at the end of the bar and travels backward. It is desirable that the mesh refinement follows the movement of the stress wave peak in the adaptive analysis. Obviously, for a spatial location **"A"** in the middle of the bar, the stress wave peak will

be passing through **"A"** twice. When the wave peak travels forward to pass **"A"** the first time, mesh refinement is provided for the spatial location **"A"** by a *fission* process. As a result, a rather accurate distribution of the material properties is computed in **"A"** on the refined mesh. After the wave leaves **"A"**, the refinement is not necessary any more and thus is removed by a *fusion* process. Therefore, the accurate distribution of the material properties that was originally computed on a refined mesh at location **"A"** will now have to be interpolated on much coarser mesh after the removal of the mesh refinement in **"A"**. Obviously, the fidelity of the solution is compromised and lost forever. When the stress wave travels backward through the spatial location **"A"** the second time, the mesh refinement is again increased by another *fission* process. But now the originally computed accurate distribution of the material properties that is interpolated on the refined mesh before the *fusion* process has been lost in the *fusion* process. The only available data is the much less accurately interpolated information from the coarser mesh.

To avoid this type of the loss of solution fidelity and to improve the *fission and fusion* data transfer scheme, a *mastergrid* scheme is proposed in this study to store the *element-wise* variables. Because the computer implementation of s-adaptive analysis is quite complicated, it is often necessary to introduce some other restrictions for reasons of practicality. For example, an *overlay mesh limit* is introduced to limit the number of layers of overlay meshes that can be applied in the adaptive analysis to make sure the scale of the composite mesh will not grow beyond the capability of the computer being

used for the analysis. When the *overlay mesh limit* is exceeded, the discretization of the global mesh and the *overlay mesh parameters* d_h and d_v are needed to be reassessed. The choice of a specific *overlay mesh limit* and a specific global mesh uniquely determine the size of the most refined composite mesh allowable in the solution process. Therefore, it is possible to create a *mastergrid* which is defined as a conventional (one-layer) mesh that has the same discretization as the most refined composite mesh allowable generated in the way that the number of the mesh layers equals the maximum number of mesh layers occurred in the course of the analysis and each layer of overlay mesh is fully superimposed. consider an adaptive analysis that begins with a 10×5 global mesh and uses the *overlay mesh parameters* $d_h=d_v=2$ in conjunction with an *overlay mesh limit* of 2 (i.e. a maximum of 2 overlay mesh levels are permitted). In this case, the *mastergrid* is a conventional 40×20 mesh, which appears exactly like the composite mesh after covering the entire 10×5 global mesh with two levels of overlay mesh. The *mastergrid* can then be used as a template on which to store the most accurate spatial distribution of element-wise variables computed, and thus preserves the solution fidelity through repeated refinement and coarsening.

With the mastergrid set aside, the *element-wise* variables of all of the top elements in the composite mesh are stored into the corresponding elements in the mastergrid after the solution is accepted at each time step. When the mesh is changed and a new composite mesh is generated, the *element-wise* variables of all the *top elements* in the new

composite mesh are obtained from the corresponding elements in the mastergrid. When a top element Ω in the composite mesh is not an element in the mastergrid, it must necessarily be a union of the elements Ω_i (i=1,2,…) in the mastergrid. In this case, a scheme similar to *fission and fusion* data transfer scheme is adopted. While saving the *element-wise* variables of element Ω into the mastergrid, all the elements Ω_i (i=1,2,…) in the mastergrid are assigned the values of the *element-wise* variables of element Ω. While restoring the *element-wise* variables of element Ω from the mastergrid, the *element-wise* variables of element Ω are taken to be the area-weighted average of all the corresponding elements Ω_i (i=1,2,…) in the mastergrid.

7.2.3 Calculation of System Matrices and Vectors

The material nonlinearity considered in this chapter will introduce some new difficulties to the calculation of the composite stiffness matrix generated in the structured mesh superposition. This section is devoted to addressing the issue.

As revealed in Chapter 3, the composite stiffness matrix possesses hierarchical structure, which is embodied in the existence of global stiffness matrix, local stiffness matrix and coupling stiffness matrix. The governing finite element system of equations of an overlay element Ω_e, which is superimposed on a global element Ω_E, can be represented in the following form:

$$\begin{bmatrix} \mathbf{K}_e^{GG} & \mathbf{K}_e^{GL} \\ \mathbf{K}_e^{LG} & \mathbf{K}_e^{LL} \end{bmatrix} \begin{Bmatrix} \mathbf{d}_E^G \\ \mathbf{d}_e^L \end{Bmatrix} = \begin{Bmatrix} \mathbf{f}_e^G \\ \mathbf{f}_e^L \end{Bmatrix} \tag{7.15}$$

where the matrices $\mathbf{K}_e^{GG}, \mathbf{K}_e^{LL}$ are the element stiffness matrices corresponding to the global degrees of freedom $\mathbf{d}_E^G$ of the global element Ω_E, and the local degrees of freedom $\mathbf{d}_e^L$ of the overlay element Ω_e, respectively. The matrix $\mathbf{K}_e^{LG} = (\mathbf{K}_e^{GL})^T$ represents a coupling matrix between the global and local degrees of freedom. The vectors $\mathbf{f}_e^G$ and $\mathbf{f}_e^L$ represent the element force vectors corresponding to the global and local degrees of freedom, respectively. The above matrices and vectors take the following forms:

$$\mathbf{K}_e^{GG} = \int_{\Omega_e} (\mathbf{B}^G)^T \mathbf{D}\mathbf{B}^G d\Omega, \quad \mathbf{K}_e^{LL} = \int_{\Omega_e} (\mathbf{B}^L)^T \mathbf{D}\mathbf{B}^L d\Omega, \quad \mathbf{K}_e^{GL} = \int_{\Omega_e} (\mathbf{B}^G)^T \mathbf{D}\mathbf{B}^L d\Omega \tag{7.16a,b,c}$$

$$\mathbf{f}_e^G = \int_{\Omega_e} (\mathbf{N}^G)^T \mathbf{f} d\Omega + \int_{S_e^\sigma} (\mathbf{N}^G)^T \mathbf{T} dS, \quad \mathbf{f}_e^G = \int_{\Omega_e} (\mathbf{N}^L)^T \mathbf{f} d\Omega + \int_{S_e^\sigma} (\mathbf{N}^L)^T \mathbf{T} dS \tag{7.17a,b}$$

where Ω_e represents the overlay element domain, S_e^σ represents the element surface where the external forces are applied, **u** is the displacement vector, **D** is the elasticity matrix, **f** is the body force vector and **T** is the surface force vector.

7.2.3.1 Calculation of Global Stiffness Matrix

As mentioned in section 7.1.1, the damage in a 2-D quadrilateral element is assumed to be uniformly distributed in the element and is represented by the damage eigenvalues evaluated at the single reduced integration point within the element.

In the conventional FEM, there is no overlay mesh superimposed on the global mesh. Thus, the only mesh is the global mesh and the only elements are the global elements, Ω_E. All the functions in the element stiffness matrix $\mathbf{K}_E^{GG}$ of a *global element* Ω_E are defined in the same coordinate system of the global element. Thus, $\mathbf{K}_E^{GG}$ can be calculated standardly using equation (7.18), where the elasticity matrix **D** is a function of the element and determined by the element's undamaged/virgin material properties and the damage eigenvalues. The global stiffness matrix $\mathbf{K}^{GG}$ is then assembled from all the element stiffness matrices like $\mathbf{K}_E^{GG}$ of element Ω_E.

$$\mathbf{K}_E^{GG} = \int_{\Omega_E} (\mathbf{B}^G)^T \mathbf{D} \mathbf{B}^G \mathrm{d}\Omega = \int_{\Omega_E} (\mathbf{B}^G)^T \mathbf{D}\left(\Omega_E\right) \mathbf{B}^G \mathrm{d}\Omega \tag{7.18}$$

However, in s-refinement, the mesh superposition process generates a composite mesh, which contains *top elements* and *non-top elements*. All *top elements* form a seamless and non-overlapping *top mesh* fully covering the problem domain. The total displacement field and the true damage field are fully defined in the *top mesh*. The non-uniformly distributed damage field on the problem domain is characterized by piecewise uniform distributions of damage in the top elements. Now, let us consider a global element Ω_E which contains several top elements Ω_e, or, simply $\Omega_E = \bigcup_e \Omega_e$. Of course, the damage eigenvalues for the global element can be obtained from its top elements by some kind of averaging scheme, such as area-weighted averaging. However, if this averaged eigenvalue is taken to represent the actual damage distribution in the global element, in essence, the averaged uniform damage distribution is used to replace the piecewise

uniform distribution of damage in the global element. Obviously, error is introduced in the averaging process. In order to accurately represent the piecewise uniform damage distribution in the global element Ω_E, the element global stiffness matrix $\mathbf{K}_E^{GG}$ of Ω_E will have to be integrated piecewise on its top elements using

$$\mathbf{K}_E^{GG} = \sum_e \int_{\Omega_e} (\mathbf{B}^G)^T \mathbf{D}\mathbf{B}^G d\Omega \tag{7.19}$$

Note that in equation (7.19), the piecewise integration domains are on top elements Ω_e^L, while the strain shape functions $\mathbf{B}^G$ are defined in the coordinate systems of the global element Ω_E. Therefore, the *global-local transformation matrix* **C** introduced in section 3.4 is utilized to facilitate the piecewise integration on the top elements.

It has been described in section 3.4 that the *global-local transformation matrix* **C** can be used to express the interpolation functions of the global element in terms of the interpolations functions of the local element. By using matrix **C**, the strain shape function of the global element can be expressed as

$$\mathbf{B}^G = \mathbf{L}\mathbf{N}^G = \mathbf{L}(\mathbf{N}^L\mathbf{C}^T) = (\mathbf{L}\mathbf{N}^L)\mathbf{C}^T = \mathbf{B}^L\mathbf{C}^T \tag{7.20}$$

Therefore, the contribution of the top element Ω_e to the element stiffness matrix $\mathbf{K}_E^{GG}$ of the global element Ω_E in equation (7.19) can be obtained as

$$\begin{aligned}\int_{\Omega_e} (\mathbf{B}^G)^T \mathbf{D}\mathbf{B}^G d\Omega &= \int_{\Omega_e} [\mathbf{B}^L\mathbf{C}^T]^T \mathbf{D}[\mathbf{B}^L\mathbf{C}^T] d\Omega = \int_{\Omega_e} \mathbf{C}[\mathbf{B}^L]^T \mathbf{D}\mathbf{B}^L\mathbf{C}^T d\Omega \\ &= \mathbf{C}\left[\int_{\Omega_e} [\mathbf{B}^L]^T \mathbf{D}\mathbf{B}^L d\Omega\right]\mathbf{C}^T = \mathbf{C}\mathbf{K}_e^{LL}\mathbf{C}^T\end{aligned} \tag{7.21}$$

Therefore, once the element stiffness matrix $\mathbf{K}_e^{LL}$ is formed for the top element Ω_e by standard integration over Ω_e, the contribution of the top element Ω_e to $\mathbf{K}_E^{GG}$ can be easily obtained by left-multiplying and right-multiplying $\mathbf{K}_e^{LL}$ with the corresponding global-local transformation matrix $\mathbf{C}$ and $\mathbf{C}^T$. By summing up all the contributions of the top elements to $\mathbf{K}_E^{GG}$ using equation (7.19), $\mathbf{K}_E^{GG}$ is able to accurately represent the piecewise uniform damage distribution within Ω_E.

7.2.3.2 Calculation of Coupling Stiffness Matrix

The calculation of the element coupling stiffness matrix in the nonlinear progressive damage problems is performed in the same manner used in linear problems. The element coupling stiffness matrix between a global element Ω_E and a local element Ω_e is calculated using the element global stiffness matrix of Ω_e and the global-local transformation matrix $\mathbf{C}$ as in section 3.4. Note that the local element Ω_e may not be a top element in multi-level mesh superposition, and in this case, its global element stiffness matrix shall be obtained as described in above.

7.2.3.3 Calculation of Internal Force Vector

It is known that in the finite element analysis of the nonlinear problems, the residual force vector is driving the problem. The residual force vector is defined as the difference between the external force vector (representing the externally applied forces) and the

internal force vector (representing the element stresses). The solution is iterated until the residual force vector is approximately zero. In the conventional FEM, the internal force vector $\mathbf{f}^{int}$ is assembled from the element internal force vector $\mathbf{f}_e$ which is given by

$$\mathbf{f}_e = \int_{\Omega_e} \mathbf{B}^T \boldsymbol{\sigma} d\Omega \tag{7.22}$$

where $\mathbf{B}$ is the strain shape function matrix, $\boldsymbol{\sigma}$ is the stress vector and Ω_e is the element domain.

There are two types of elements in the composite mesh, the top elements and the non-top elements. The element internal force vectors of the top elements and the element internal force vectors of the non-top elements are all assembled to form the internal force vector for the composite mesh. For the top elements, their element internal force vectors are calculated using equation (7.22) exactly like in the convention FEM. However, for non-top elements, their element internal force vectors are calculated differently in order to accurately represent the piecewise uniform damage distribution in their element domains. Similarly, consider a non-top element Ω_E which contains several top elements Ω_e as $\Omega_E = \bigcup_e \Omega_e$, the element internal force vector $\mathbf{f}_E$ will have to be integrated piecewise on its top elements using

$$\mathbf{f}_E = \sum_e \int_{\Omega_e} (\mathbf{B}^G)^T \boldsymbol{\sigma} \mathrm{d}\Omega \tag{7.23}$$

Similarly, with the help of the global-local transformation matrix **C**, the contribution of the top element Ω_e to the element internal force vector $\mathbf{f}_E$ of the non-top element Ω_E in equation (7.23) can be obtained as

$$\int_{\Omega_e} (\mathbf{B}^G)^T \boldsymbol{\sigma} d\Omega = \int_{\Omega_e} [\mathbf{B}^L \mathbf{C}^T]^T \boldsymbol{\sigma} d\Omega = \int_{\Omega_e} \mathbf{C}[\mathbf{B}^L]^T \boldsymbol{\sigma} d\Omega = \mathbf{C} \int_{\Omega_e} [\mathbf{B}^L]^T \boldsymbol{\sigma} d\Omega = \mathbf{C}\mathbf{f}_e \quad (7.24)$$

Once the element internal force vector $\mathbf{f}_e$ is formed for the top element Ω_e by standard integration over Ω_e, the contribution of the top element Ω_e to $\mathbf{f}_E$ can be easily obtained by multiplying $\mathbf{f}_e$ with the corresponding global-local transformation matrix **C**. By summing up all the contributions of the top elements to $\mathbf{f}_E$ using equation (7.23), $\mathbf{f}_E$ is able to accurately represent the piecewise uniform damage distribution within Ω_E.

7.3 Numerical Examples

Five 2-D example problems are presented to demonstrate the performance characteristics of the structured, multilevel, s-adaptive procedure in nonlinear problems caused by progressive damage. The first two problems are quasi-static problems and the last three problems are transient problems. The 4-node linear quadrilateral elements are used in all five the examples.

7.3.1 Uniaxial Tension Test

A simple quasi-static uniaxial tensile test specimen is simulated with a 2-D plane stress model to demonstrate the progressive damage behavior of a typical material in a familiar setting. This simple problem is characterized by a state of homogeneous deformation and loading, consequently the problem can be solved with a single element, without the need for adaptivity of any kind. The initial undamaged isotropic material properties of the specimen are $E_0 = 100$, $\nu_0 = 0.3$. The damage surface is defined as $F(\mathbf{Y},B) = (1/2)[J_{11}(Y_1)^2 + J_{22}(Y_2)^2] - (B_0 + B)$, where $J_{11} = J_{22} = 6$ and linear damage hardening are $B_0 = 0$, $B = c_1\beta = 0.001\beta$. In the test, the maximum imposed axial stress of $\boldsymbol{\sigma}_{max} = 1.2240$ was applied via a series of 500 equal stress increments. The *convergence check* in Newton-Raphson iterations requires all components of the residual force vector to be less than 10^{-10}.

The uniaxial loading is applied in the y direction; therefore, $\sigma_2>0$ and $\sigma_1=\sigma_3=0$, consequently, the damage eigenvalue D_2 attains a finite positive value while D_1 remains zero.

Figure 7.2 shows the computed nonlinear relationship between the axial stress component σ_2, the axial damage eigenvalue D_2 and the axial strain component ε_2. The tensile test is stopped when the applied stress reaches $\boldsymbol{\sigma}_{max} = 1.2240$. This maximum stress value is purposefully chosen since it occurs just prior to when the stress/strain relationship

becomes horizontal caused by damage-induced softening. In other words, σ_{max} = 1.2240 is the maximum load the specimen can bear without completely rupturing. If the specimen is loaded beyond σ_{max} = 1.2240, the damaged state of the material will not permit the internal material forces to be further increased, hence axial force equilibrium can not be achieved. In this case, the magnitude of the damage eigenvalue D_2 quickly escalates toward D_2=1 while the axial displacement increments increase without bound, indicating a complete material failure.

In Figure 7.2, the damage eigenvalue D_2 begins evolving at the start of the loading, but it does not achieve a visually significant value until the axial strain has reached approximately 0.6%. Similarly, the stress/strain relationship does not show significant nonlinearity until D_2 reaches approximately 2%.

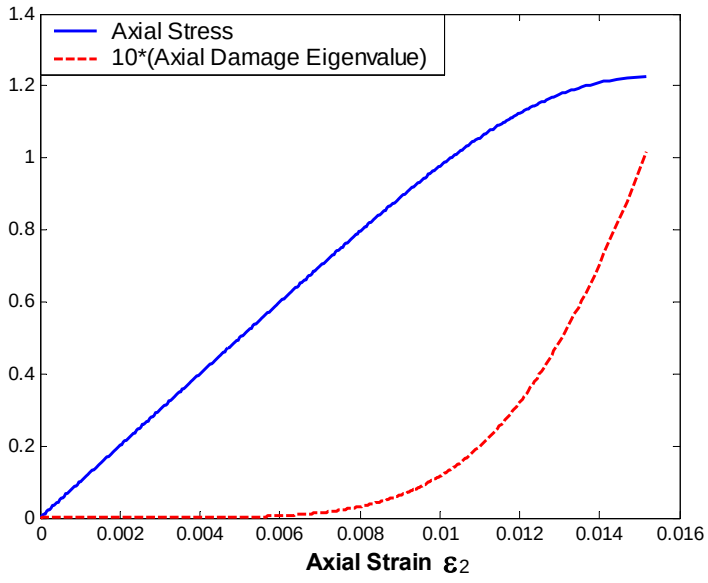


Figure 7.2: Relationship between σ_2, D_2 and ε_2

Figure 7.3 shows the damaged material modulus E_{22} in y direction due to the presence of D_2. The damaged E_{22} is reduced to approximately 80% of the original undamaged/virgin value at the concluding point.

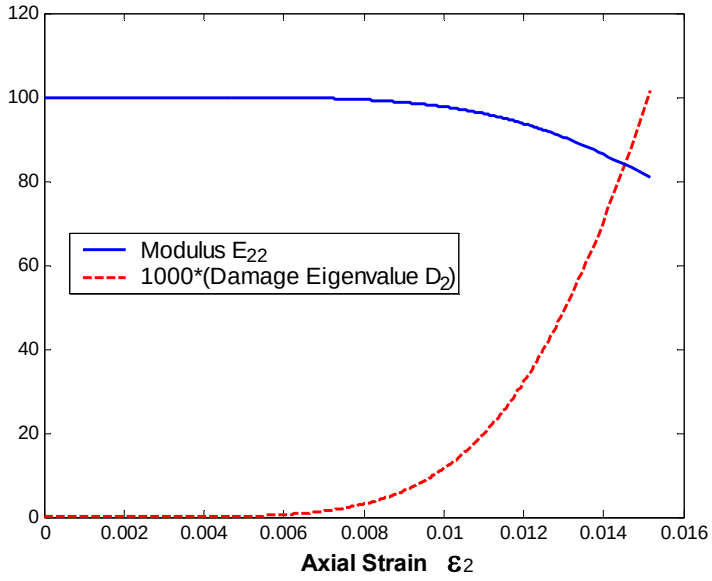


Figure 7.3: Relationship between E_{22} , D_2 and ε_2

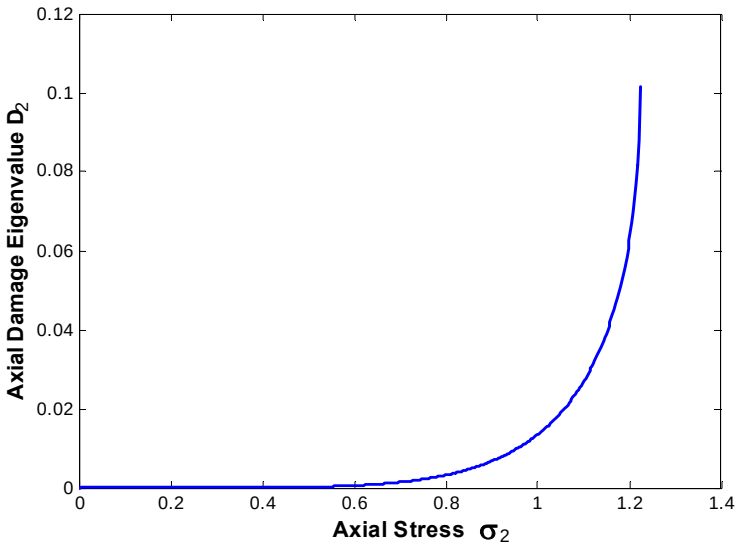


Figure 7.4: Relationship between D_2 and σ_2

Figure 7.4 shows the relationship between σ_2 and D_2. It is observed that the damage increases exponentially as the applied stress approaches the concluding point. In the vicinity of the concluding point, a slight increase in applied stress may cause a big increase in damage. Therefore, it is concluded that the damage is very sensitive to the external load once the damage has accumulated to a certain extent. This characteristic of damage plays a decisive role in causing the oscillations of the damage distribution in the adaptive analysis, which will be shown in the transient examples in this section.

7.3.2 Infinite Plate with a Circular Hole under Uniform Load

Now let us consider a progressive damage problem that exhibits spatially varying stress and strain fields in a quasi-static setting. This permits the spatial adaptivity of the structured s-adaptive method to be demonstrated without the complicating effect of transient behavior. The classical stress concentration problem of an infinite plate with a circular hole is chosen to demonstrate the performance of the s-adaptive method in resolving the progressive damage behavior associated with localized stress concentrations. The geometry, material properties and external loading are given in Figure 7.5(a).

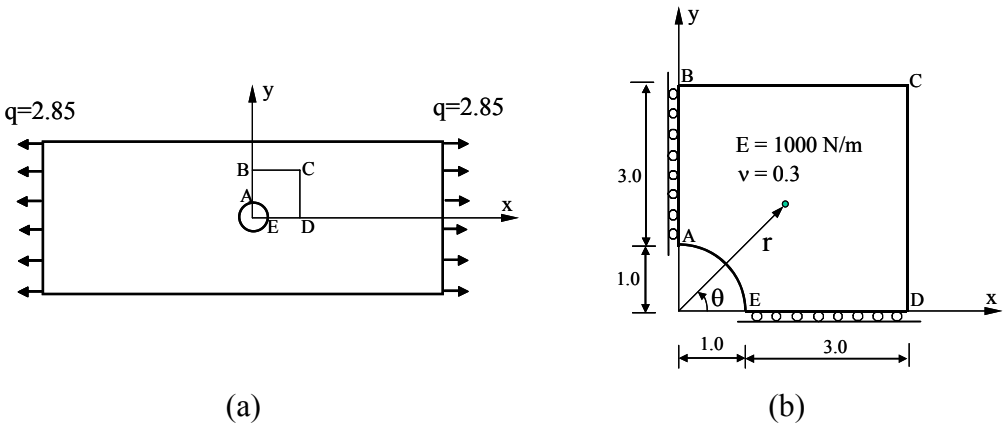


Figure 7.5: Infinite plate with a circular hole under uniform tensile load

It is known that the linear elastic problem has a theoretical stress solution given by equations (7.25a,b,c). In this study, the problem is solved as a nonlinear problem caused by progressive damage and the stress distribution in Equation 7.25 serves as traction boundary conditions on edges *BC* and *CD* of the truncated computational domain shown in Figure 7.5(b). Symmetric displacement boundary conditions are specified on edges *AB* and *ED*. A state of plane stress is assumed. The externally applied traction q=2.85 is imposed via 20 equal load increments. The initial undamaged isotropic material properties are E = 1000, ν = 0.3. The damage surface is defined as $F(\mathbf{Y},B) = (1/2)[J_{11}(Y_1)^2 + J_{22}(Y_2)^2] - (B_0 + B)$, where the damage resistance constants are $J_{11} = J_{22} = 1$ and the linear damage hardening are $B_0 = 0$, $B = c_1\beta = 0.001\beta$. The *convergence check* in the Newton-Raphson iterations requires that all components of the residual force vector to be less than 10^{-10}. The spatial error tolerance is chosen as η_s^{TOL}=2.5% and the safety factor is $\theta_s = 0.6$. The *overlay mesh parameters* are chosen as $d_h = d_v = 2$. The

geometry and the loading of the problem determine that the predominant damage occurs in x-direction and the maximum damage occurs in the location of point **A** in Figure 7.5.

$$\sigma_x = q\left(1-\frac{r_0^2}{r^2}\left(\frac{3}{2}\cos 2\theta+\cos 4\theta\right)+\frac{3}{2}\frac{r_0^4}{r^4}\cos 4\theta\right) \tag{7.25a}$$

$$\sigma_y = q\left(-\frac{r_0^2}{r^2}\left(\frac{1}{2}\cos 2\theta-\cos 4\theta\right)-\frac{3}{2}\frac{r_0^4}{r^4}\cos 4\theta\right) \tag{7.25b}$$

$$\tau_{xy} = q\left(-\frac{r_0^2}{r^2}\left(\frac{1}{2}\sin 2\theta+\sin 4\theta\right)+\frac{3}{2}\frac{r_0^4}{r^4}\sin 4\theta\right) \tag{7.25c}$$

where (r,θ) are the polar coordinates and $r_0 = 1$ is the radius of the hole.

Figure 7.6 shows the original 2-D global mesh (labeled as mesh 1) chosen for the problem, along with a series of four composite meshes (meshes 2 through 5) that are generated by s-adaptive analysis. In the s-adaptive analysis, new levels of overlay meshes are added locally to the existing elements of the composite mesh until the desired spatial error tolerance is met throughout the computational domain. For the chosen spatial error tolerance, a total of four levels of overlay meshes are required. Note that the first level of overlay mesh in the composite mesh (mesh 2) completely covers the whole domain, since the original global mesh was quite coarse and hence unable to achieve the spatial error tolerance in any part of the problem domain. In deed, one of the major advantages of the adaptive methodology is that the user is not required to guess or intuit an adequate original global mesh; instead, the adaptive methodology automatically constructs the optimal composite mesh regardless of the starting level of discretization. Also seen in

Figure 7.6 is the well-anticipated trend that the successive layers of overlay meshes tend to become concentrated in the critical area of the stress concentration in the immediate vicinity of the hole.

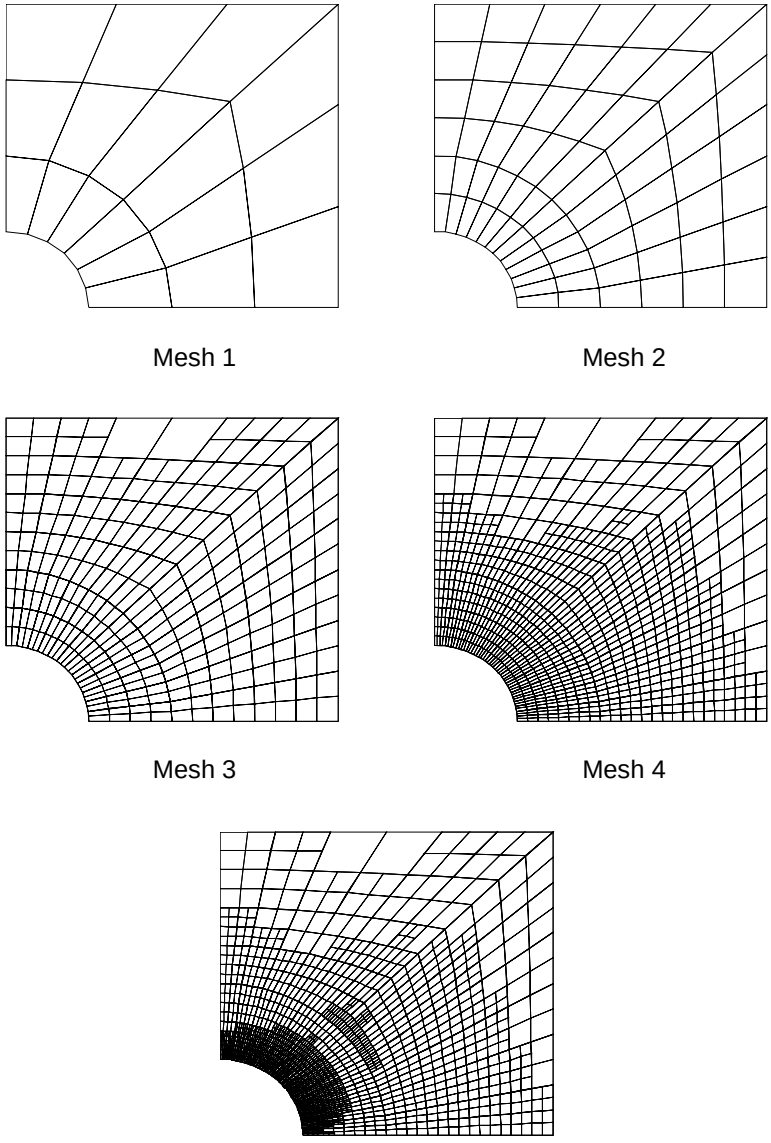


Figure 7.6: Series of composite meshes generated by s-adaptive analysis

It is worth mentioning that the same rank deficiency issue in SPR procedure as described in Chapter 4 is encountered in this analysis while calculating the postprocessed stresses with the use of 4-node quadrilateral elements. The proposed *rotated* SPR procedure is utilized in all the composite meshes in Figure 7.6 to overcome the rank deficiency issue.

For comparison, the same infinite plate problem is also solved using a series of uniformly refined conventional 2-D meshes, each of which corresponds to a composite mesh in Figure 7.6 and looks like the composite mesh in Figure 7.6 except that it is fully superimposed instead of being partially superimposed. In fact, the first and second uniform refined meshes look exactly like mesh 1 and mesh 2 in Figure 7.6. Table 7.1 shows the comparisons between the results of the series of composite meshes and their corresponding uniform refined meshes. Each composite mesh has two rows of data; the top row lists the results of the composite mesh and the bottom row lists the results of its corresponding uniform refined mesh. For each composite, the results include the maximum damage eigenvalue D_1 (the damage eigenvalue in x-direction) in the current composite mesh (listed in column *Max. Damage*); the x-stress in the element where the maximum damage occurs (listed in column σ_x); the ratio of the damaged material module E_{11} to its original undamaged value in the element where the maximum damage occurs (listed in column *MRM*); the total strain energy stored in the truncated computational domain (listed in column *SE*); the estimated percentage spatial error (listed in column *PSE*); the number of active degrees of freedom (listed in column *ADOF*).

Table 7.1: Comparisons of the solution results by the composite meshes and their corresponding uniform refined meshes

	Max. Damage	σ_x (N/m^2)	SE (J)	MRM (%)	ADOF	PSE (%)
Mesh 1	0.0026848	4.56163	0.116270	99.46	48	12.13840
	0.0026848	4.56163	0.116270	99.46	48	12.13840
Mesh 2	0.0105169	5.65156	0.1168732	97.91	168	7.70160
	0.0105169	5.65156	0.1168732	97.91	168	7.70160
Mesh 3	0.0323470	6.55955	0.1171377	93.64	606	4.98148
	0.0324601	6.56232	0.1171558	93.61	624	5.04726
Mesh 4	0.0683126	6.99870	0.1172390	86.80	1776	2.92432
	0.0691297	7.00334	0.1172633	86.65	2400	3.03656
Mesh 5	0.1036717	7.07315	0.1172682	80.34	3034	2.30225
	0.1066603	7.07127	0.1172972	79.81	9408	2.10150

It is observed in Table 7.1 that the composite meshes are able to achieve almost the same solutions as the corresponding uniform refined meshes in almost every category. However, the adaptive solutions are achieved with fewer degrees of freedom than the uniform refined non-adaptive solutions. For example, the adaptive composite mesh 5 uses only 3034 active DOF, but the uniform refined mesh uses 9408 active DOF. With the superior performance in this mild stress concentration problem of infinite plate with a circular hole, the s-method is fully expected to provide even better performance in more severe stress concentration problems.

The results shown in Table 7.1 primarily compare the adaptive and non-adaptive solutions at the most severely stressed point in the problem domain. In order to provide

a more 'full-field' comparison of the solutions over the entire computational domain, Figures 7.7 through 7.14 show the predicted spatial distributions of various solution parameters. Figures (7.7-7.10) show the comparisons of the displacement distributions on edges *BC* and *CD* between the adaptive analysis and the uniform refined analysis. Those comparisons provide indicators of the performance of the s-method in accurately simulating the equivalent global stiffness of the structure. Figures (7.11-7.14) show the comparisons of the damage distributions on arc *AE* and edge *AB* between the adaptive analysis and the uniform refined analysis. These figures demonstrate that the adaptive and non-adaptive solutions do indeed show excellent agreement throughout the entire problem domain.

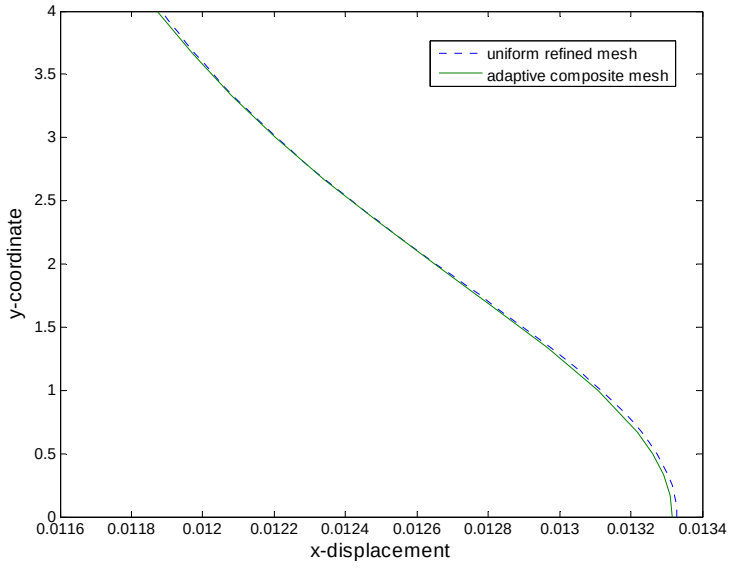


Figure 7.7: X-displacement distribution on edge CD

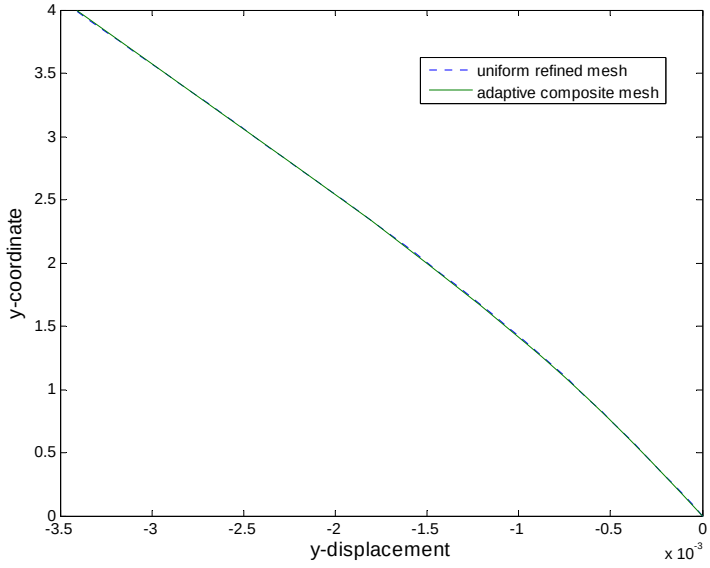


Figure 7.8: Y-displacement distribution on edge CD

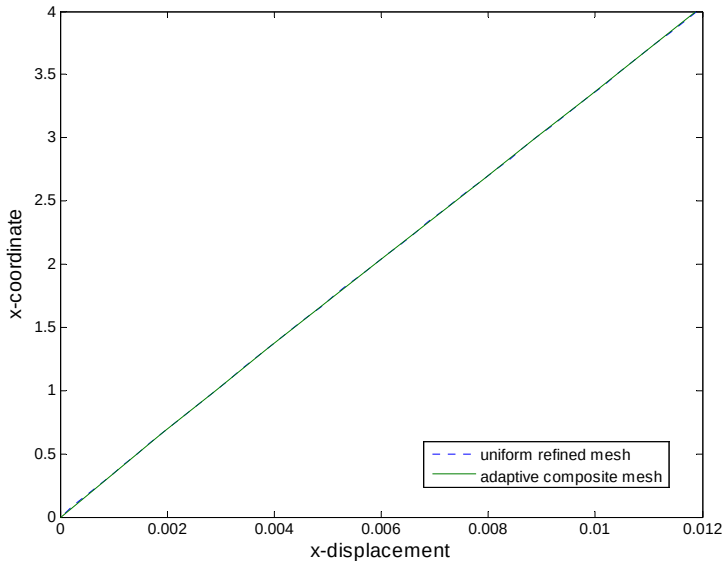


Figure 7.9: X-displacement distribution on edge BC

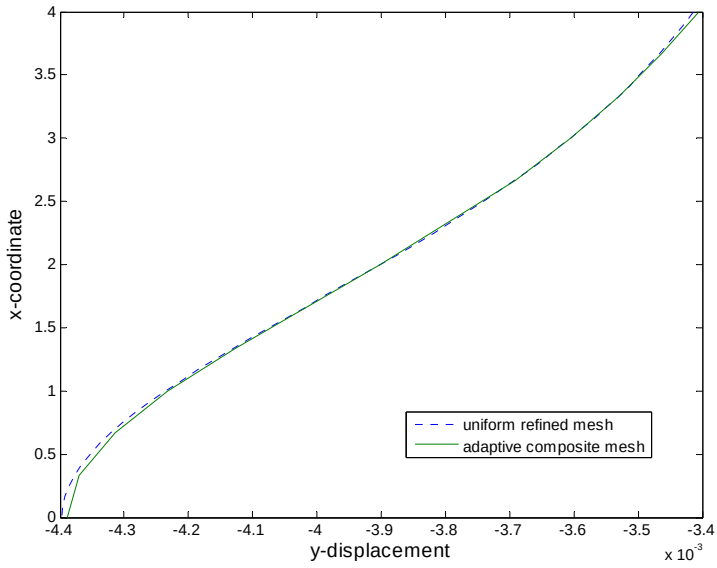


Figure 7.10: Y-displacement distribution on edge BC

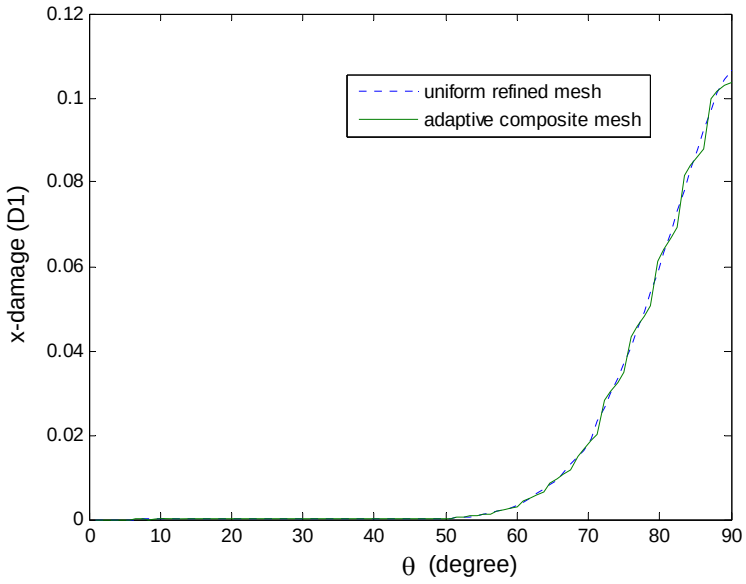


Figure 7.11: Distribution of damage eigenvalue D_1 on arc AE

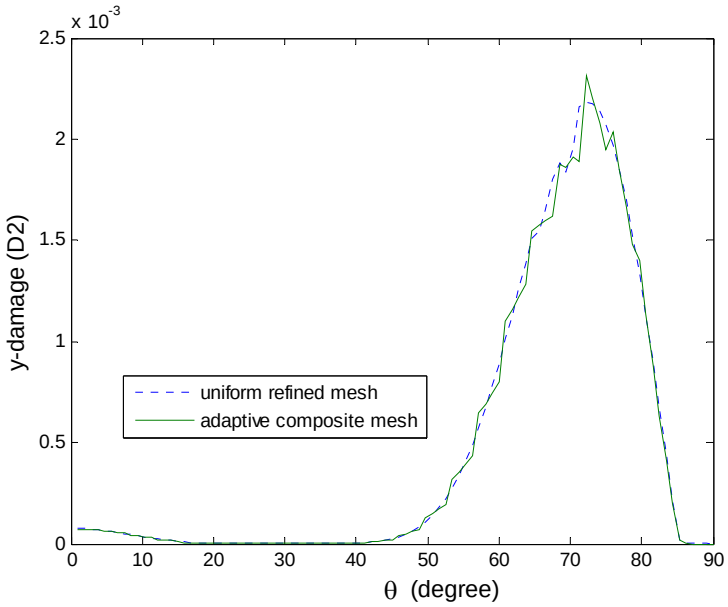


Figure 7.12: Distribution of damage eigenvalue D_2 on arc AE

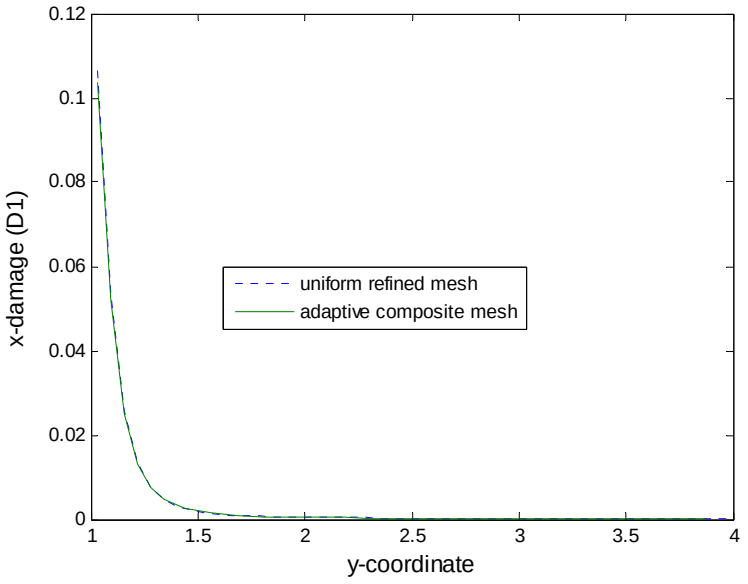


Figure 7.13: Distribution of damage eigenvalue D_1 on edge AB

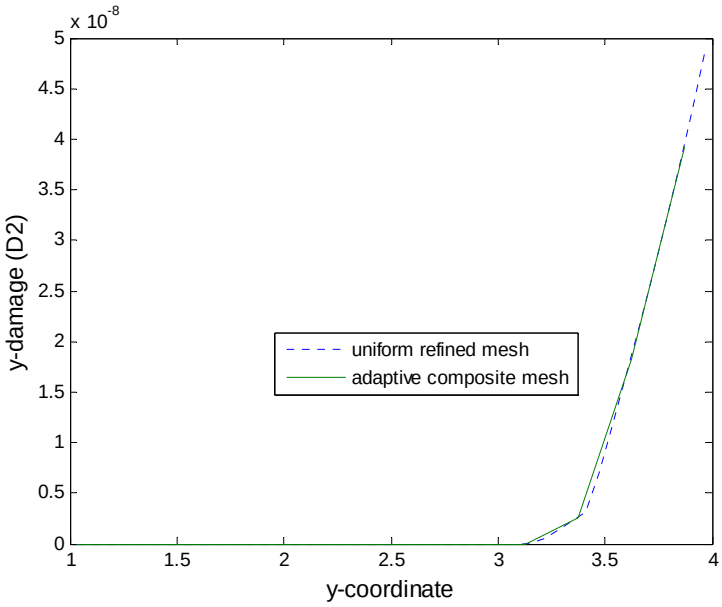


Figure 7.14: Distribution of damage eigenvalue D_2 on edge AB

7.3.3 Propagation of an Axial Stress Wave in a Rectangular Bar

Now let us shift the focus to adaptive solutions of 2-D transient problems with material nonlinearity in the form of progressive damage. The first transient problem considered is the propagation of an axial compressive wave that travels down the length of an unconstrained rectangular bar and reflects off the free end as a tensile stress wave. This particular problem is purposefully chosen since the stress field and strain field exhibit localized, rapid spatial variation only in the axial direction, not the transverse direction. This simplifies the scope of the initial transient example problem and allows attention to be focused on the unique aspects of the adaptive solution that are related to transient behavior.

Figure 7.15 shows the geometry, material properties, constraints and load of the problem. The externally applied compressive traction ***p***(t) is uniformly applied across the left end of the bar, but varies in time in the form of a half-sine-wave pulse. The problem is solved as a 2-D, undamped, nonlinear plane stress problem with material nonlinearity in the form of progressive damage. The damage surface is defined as $F(\mathbf{Y},B) = (1/2)[J_{11}(Y_1)^2 + J_{22}(Y_2)^2] - (B_0 + B)$, where $J_{11} = J_{22} = 6$ and linear damage hardening are $B_0 = 0$, $B = c_1\beta = 0.001\beta$.

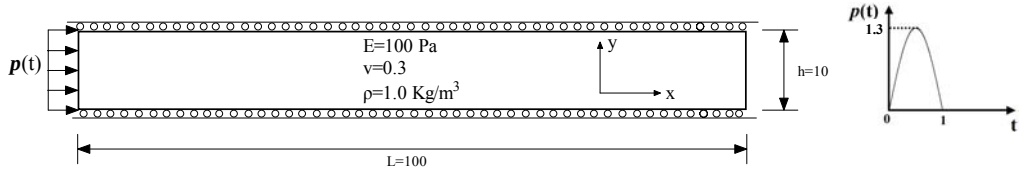


Figure 7.15: A rectangular bar under a half-sine-wave compressive pulse

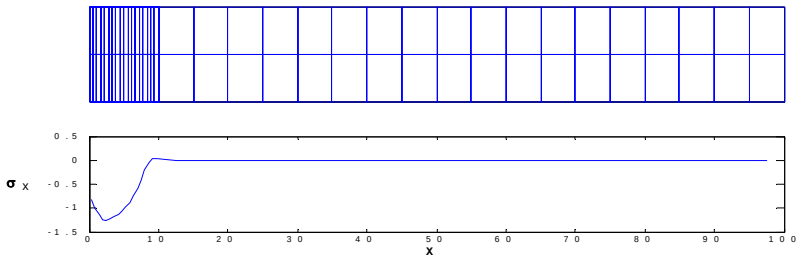
As the stress wave propagates along the rectangular bar, it causes damage. Since the dominant stress component in this problem is the stress in x direction, the dominant damage is in x-direction too. The s-adaptive solution is initiated from a uniform global mesh that contains 20 elements in the x direction and 2 elements in the y direction. Adaptivity of the structured mesh superposition solution is controlled by the following parameters: temporal error tolerance $\eta_t^{TOL} = 0.1\%$, $\beta_t = 0.6$, $\theta_t = 0.9$, $K_t = 8$, spatial error tolerance $\eta_s^{TOL} = 5\%$, $\beta_s = 0.6$, $\theta_s = 0.55$, $\phi = 0.55$, and $K_s = 8$. The overlay mesh parameters are chosen as $d_h = 3$ and $d_v = 1$. The adaptive solution is computed for time

period $0 \leq t \leq 15$ sec., which is sufficient for the stress wave to reflect off the free end and travel more than half-way back to its starting point.

Figure 7.16 shows a series of composite meshes and associated axial stress distributions that were produced at various points in time during the s-adaptive analysis. As seen in Figure 7.16, the overlay mesh is repeatedly updated to provide a composite mesh with high refinement in the immediate vicinity of the traveling stress wave, while maintaining a coarse mesh elsewhere. In order to maintain the user-specified spatial error tolerance, two levels of overlay meshes were needed, resulting in the smallest elements having a length of L/180 in the direction of wave propagation. When the compressive wave reaches the free end at the right end, it reflects as a tensile wave. When the remaining forward compressive wave and the reflected backward tensile wave meet near the free end, they offset each other and generate a superposed wave which can be adequately simulated using less refinement as shown in Figure 7.16(d).

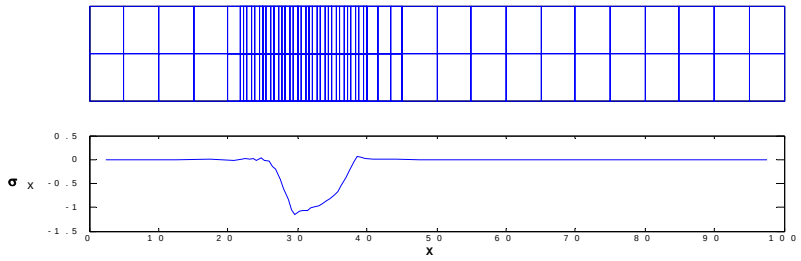
Figure 7.16: Series of composite meshes and computed axial stress distributions produced by the structured, s-adaptive solution at various points in time

a)



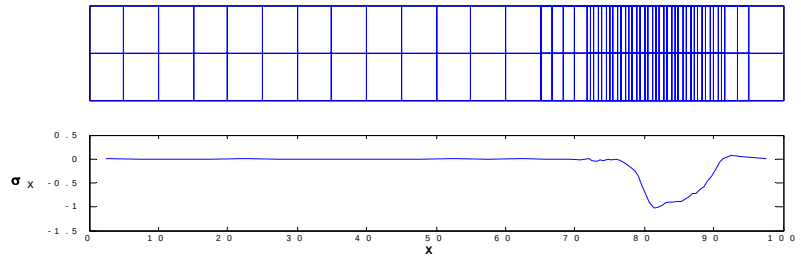
t=0.8130

b)



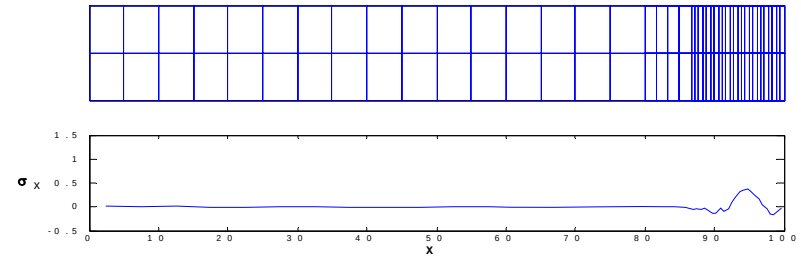
t=3.6051

c)



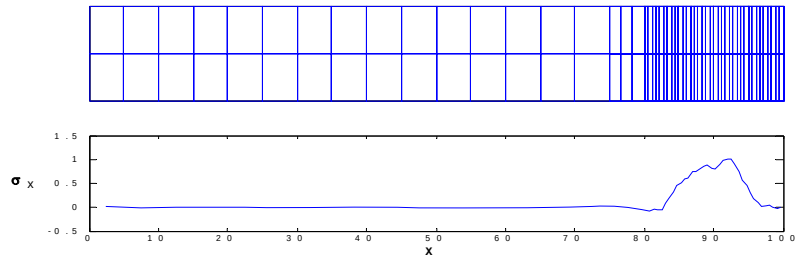
t=8.6127

d)



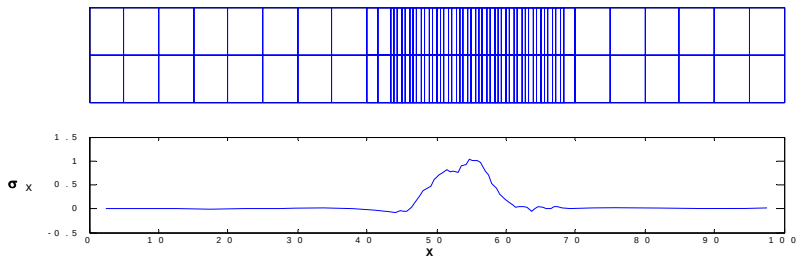
t=10.2116

e)



t=11.1014

f)



t=14.6100

Figure 7.17 shows the number of active DOF used in the composite mesh as a function of time. Note that after the stress wave becomes fully formed within the wave guide (at approximately t=1 sec.), the number of active degrees of freedom remains relatively stable, except for the time period of 9 sec. < t < 11 sec., when the stress wave reflects off the free end of the wave guide and thus less refined mesh is adequate to simulate the superposed wave near the free end. Note also that in response to the estimated spatial error, the composite mesh was updated 30 times during the course of the analysis; however, each update involves only minor changes to the overlay mesh and can be performed very quickly. The estimated relative spatial error is presented in Figure 7.18.

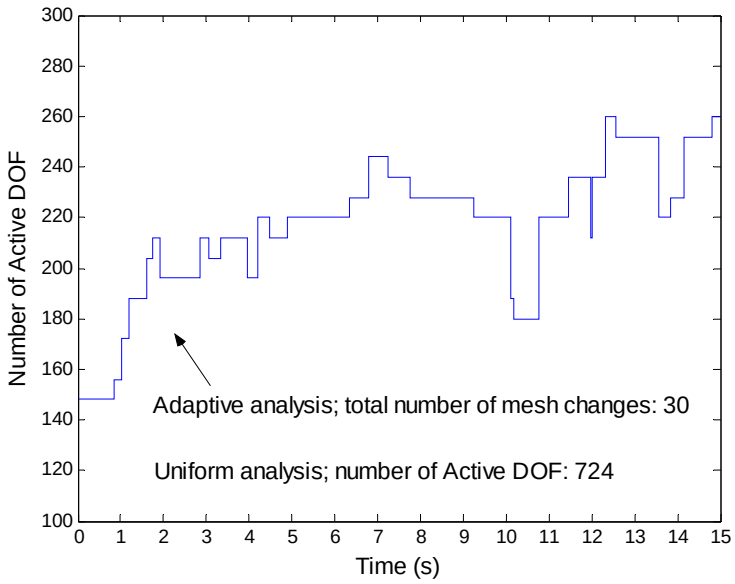


Figure 7.17: Time history of the number of active DOF used in the s-adaptive analysis

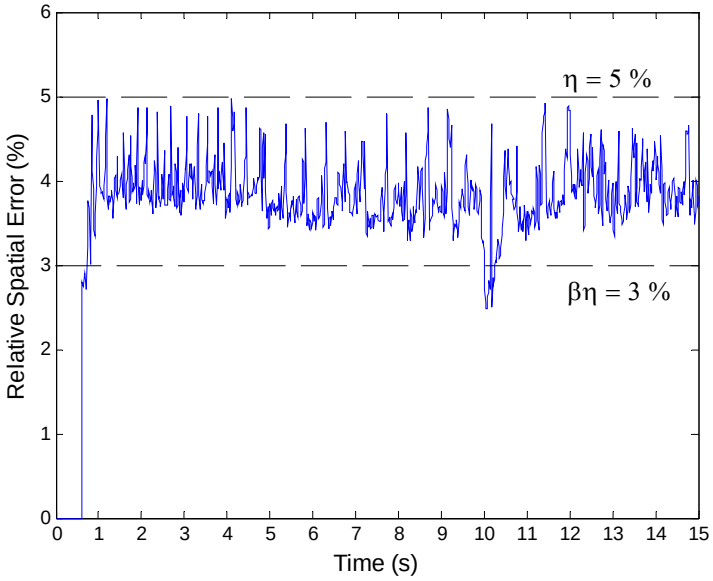


Figure 7.18: Time history of relative spatial error achieved in the adaptive analysis

The estimated temporal error is used to control the time step size over the course of the adaptive analysis. Figure 7.19 shows the time history of the time step size, while Figure 7.20 shows the estimated temporal error.

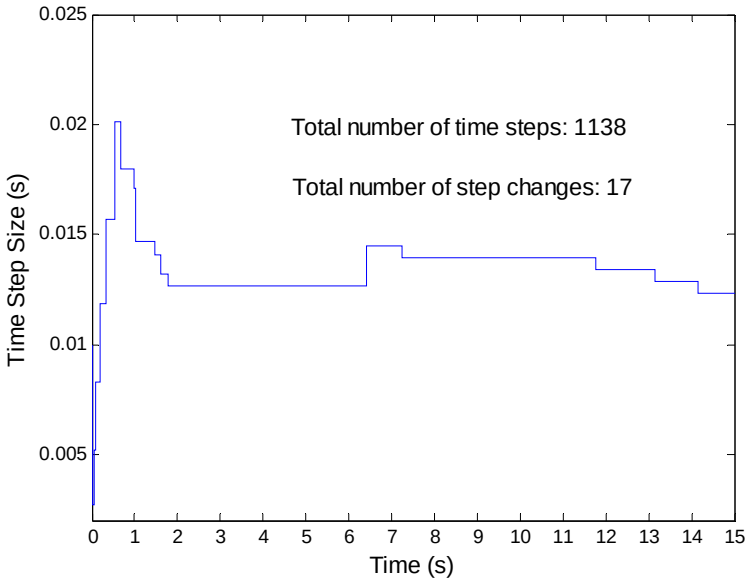


Figure 7.19: Time history of the time step size used in the adaptive analysis

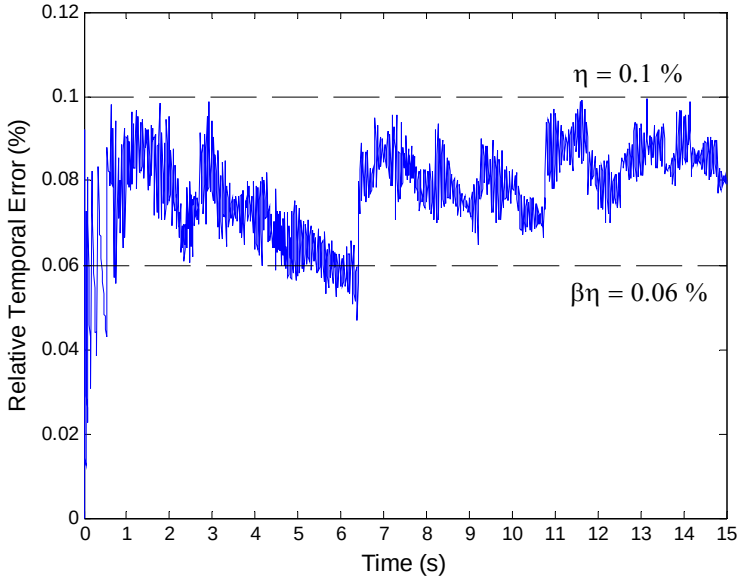


Figure 7.20: Time history of relative temporal error achieved in the adaptive analysis

For comparison, the same problem is solved with a uniform refined mesh of 180 elements in the x direction and 2 elements in the y direction (724 active DOF) with a time step size dt=0.01 sec. The uniform element size is comparable to the most refined overlay mesh level used previously in the s-adaptive solution. Figures 7.21 and 7.22 show the location, size and shape of the compressive axial stress wave predicted by both solutions at the discrete times t=1, 4, 8, 11 and 15 seconds. Despite the fact that the s-adaptive solution utilizes approximately one-third the number of active DOF used by the uniform refined solution, the two solutions show excellent agreement.

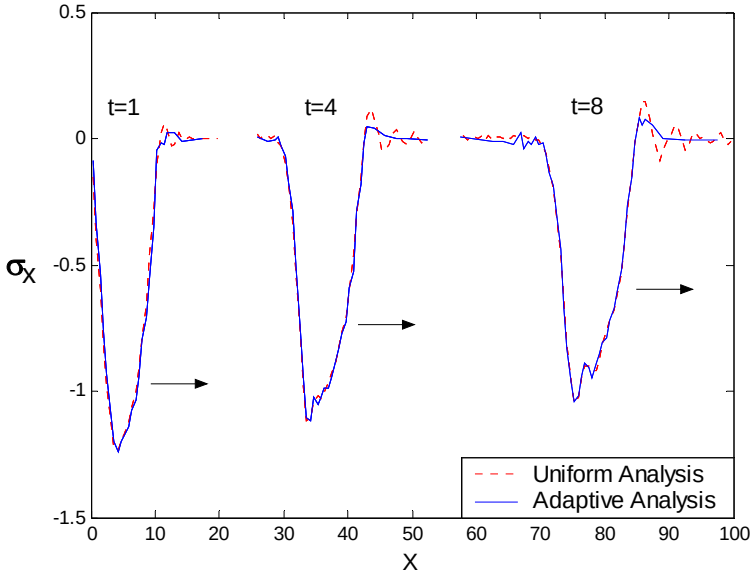


Figure 7.21: Comparison of the compressive axial stress wave predicted at discrete times t=1,4,8 sec. by the s-adaptive solution and the uniform refined solution

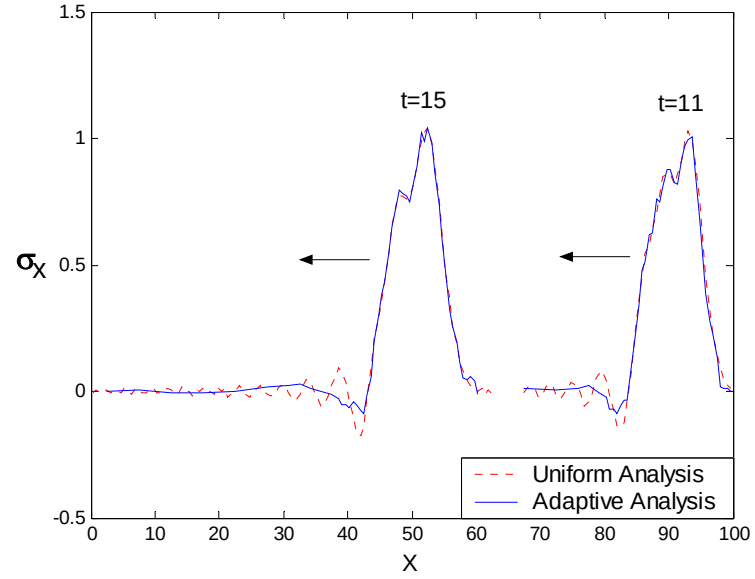


Figure 7.22: Comparison of the tensile axial stress wave predicted at discrete times t=11,15 sec. by the s-adaptive solution and the uniform refined solution

A close examination of the spurious oscillations at the leading and trailing edges of the stress wave reveals some qualitative differences between the two solution methods. Figure 7.23 shows a zoom-in view of the compressive axial stress wave at time t=5.8037 seconds. The solution obtained with the uniform refined mesh exhibits significant high frequency oscillation at the leading edge of the wave. At the trailing edge of the stress wave, the situation is reversed, i.e. the s-adaptive method exhibits more spurious oscillation than the uniform refined solution. We recall the similar scenarios occur in the linear analysis in Chapter 6. The similar explanations for the linear analysis also apply in this nonlinear analysis.

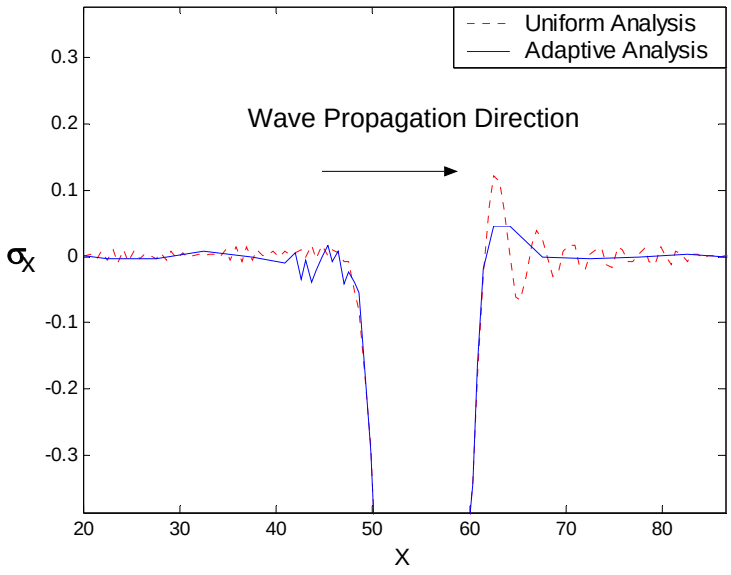


Figure 7.23: Close-up view of the compressive axial stress wave showing areas of spurious oscillation

Figure 7.24 shows the time history of the total energy (which includes the strain energy and the kinetic energy) norm of the bar. The total energy norm increases as the half sine pulse is applied; and then it begins to decrease as the stress wave propagates along the bar. The decrease part of the total energy is consumed to damage and harden the material.

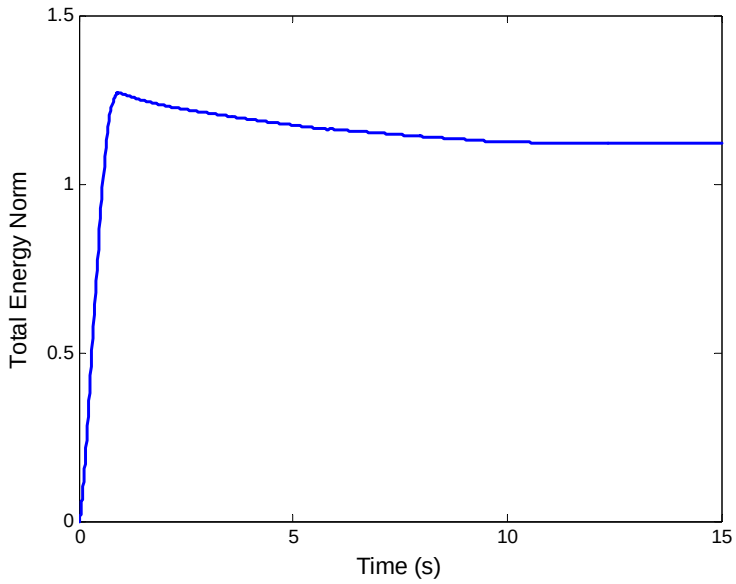


Figure 7.24: Time history of the total energy norm of the bar

Figure 7.25 shows the comparison of the damage distribution at t=15 sec. by the s-adaptive solution and the uniform refined solution. Though some noises in the adaptive solution cause the adaptive solution to oscillate above and below the uniform refined solution, the two solutions actually show good agreement. The explanation for the noise is as follows. In the s-adaptive analysis of stress propagation, the shapes of the stress waves are usually conserved very well during the mesh changes. However, the peaks or

valleys of the stress waves are also somehow affected and the oscillations are introduced at the peaks or valleys. Moreover, it is known from Figure 7.4 in the uniaxial tension test that the progressive damage is very sensitive to the magnitude of the loading once it has accumulated to a certain level. Therefore, we conclude that the oscillation in the vicinity of the stress peak or valley caused by the mesh change causes the subsequent oscillation of the damage. It is noticed in Figure 7.24 that the gradient of the decrease of the total energy is decreasing, which implies that less and less energy is consumed in damaging and hardening the material. This observation is confirmed in Figure 7.25 as the damage caused by the stress wave is decreasing as the stress wave propagates along the bar. The decrease of the total energy continues until the compressive stress wave reaches the free edge on the right. Thereafter, the total energy remains constant, which means no more damage is accumulated and the problem degenerates from a nonlinear stress wave propagation problem to a linear stress wave propagation problem.

Figure 7.26 shows the comparison of the material stiffness distribution in the bar by the s-adaptive solution and the uniform refined solution at t=15 sec. The two solutions show good agreement too.

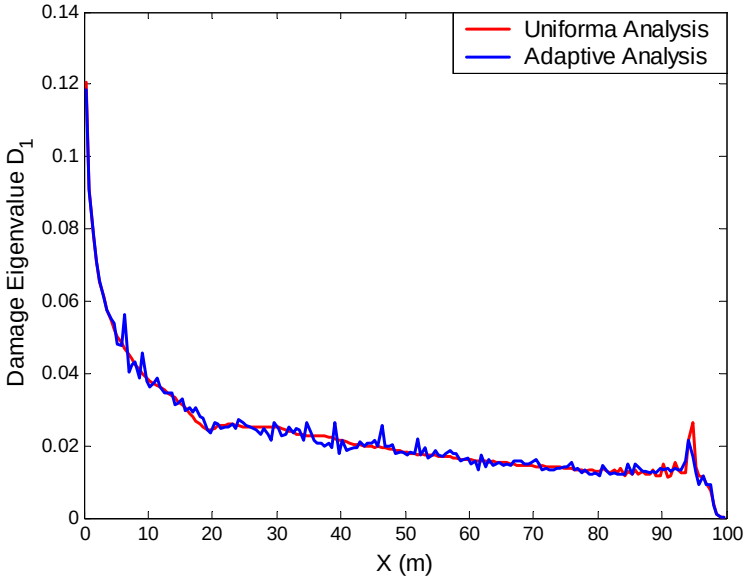


Figure 7.25: Comparison of the damage distribution in the bar by the s-adaptive solution and the uniform refined solution at t=15 sec.

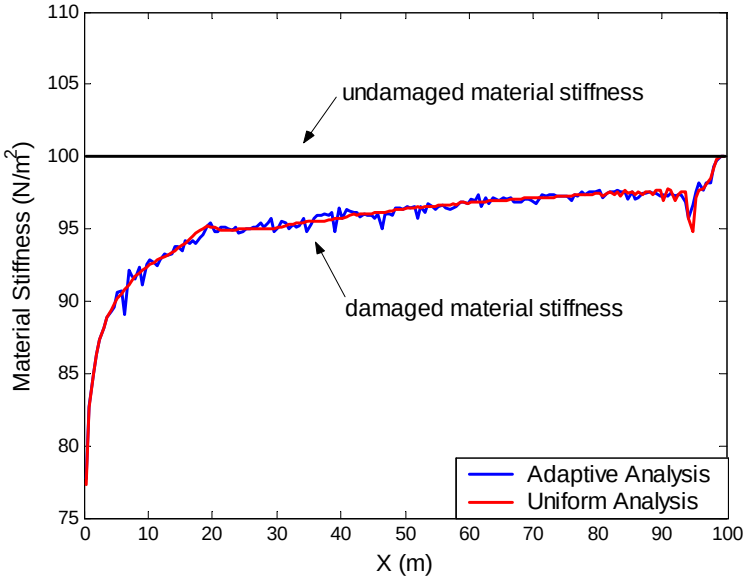


Figure 7.26: Comparison of the material stiffness distribution in the bar by the s-adaptive solution and the uniform refined solution at t=15 sec.

Figure 7.27 shows the time history of the maximum tensile stress and the maximum compressive stress in x direction within the bar. In this wave propagation problem, the maximum tensile stress occurs at the peak of the tensile stress wave and the maximum compressive stress occurs at the valley of the compressive stress wave. In Figure 7.27, it is observed that the maximum compressive stress is decreasing as the wave travels toward the free end, which is consistent with the decreasing of the total energy observed in Figure 7.24. At the same time, the maximum tensile stress remains close to zero. During the wave reflection off the free end, the compressive stress wave is transforming into the tensile stress wave, which is confirmed by the decreasing of the maximum compressive stress and the increasing of the maximum tensile stress in the period. After the complete reflection, the amplitude of the tensile stress wave remains constant that is consistent with the observation of constant total energy in Figure 7.24.

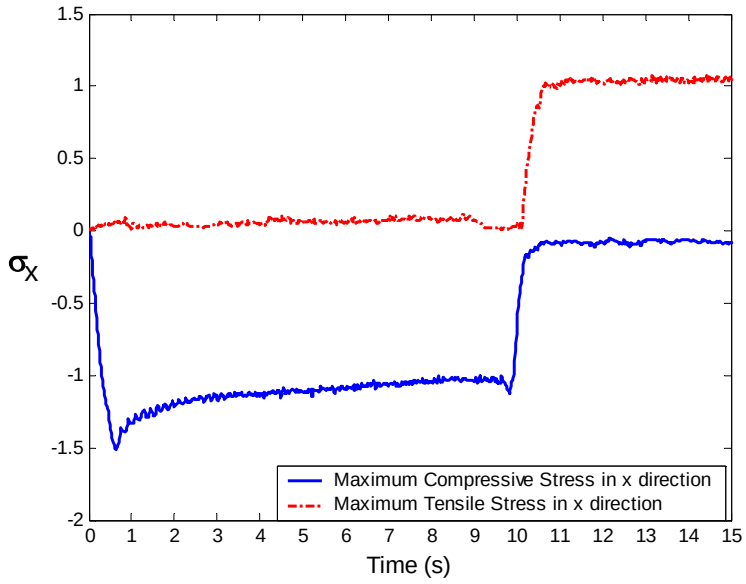


Figure 7.27: Time history of the maximum tensile and compressive stress in x direction of the bar

7.3.4 Propagation of an Axial Stress Wave in a Split Hopkinson Pressure Bar

A Split Hopkinson Pressure Bar (SHPB) is an inexpensive device to determine the dynamic properties of materials [83]. A conventional SHPB consists of a striker bar, an incident bar and a transmitter bar. The specimen is sandwiched between the transmitter and incident bar. The striker bar strikes the incident bar and generates a compressive wave (incident wave) in the incident bar. At the incident bar/specimen interface, part of the incident wave is reflected back into the incident bar as a tensile wave and the other part of the incident wave is transmitted into the specimen. When the transmitted wave passes through the specimen and reaches the transmitter bar/specimen interface, part of it is reflected back into the specimen and the other part of it is transmitted into the transmitter bar. Strain gages are mounted on the incident bar, transmitter bar and the specimen, respectively. Using the recorded incident, transmitted and reflected waves, the strain, strain rate and stress-time histories of the specimen can be derived.

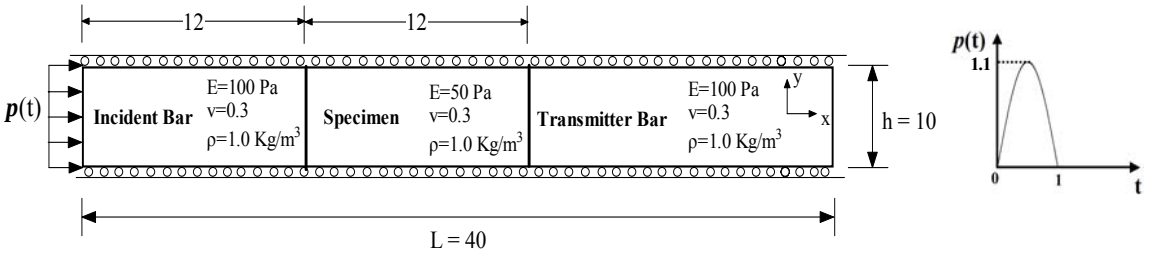


Figure 7.28: A simplified SHPB device under a half-sine-wave compressive pulse

Figure 7.28 shows a simplified model of the SHPB. The striker bar is replaced by a compressive pulse $\boldsymbol{p}(t)$ which is uniform with respect to spatial position, but varies with respect to time in the form of a half-sine-wave pulse. The geometry, material properties and constraints of the problem are also given in Figure 7.28. The length of the specimen is chosen so that it is long enough to contain a complete waveform. In the real SHPB device, the incident bar and the transmitter bar are much longer than the specimen. However, since the s-adaptive methodology has successfully demonstrated the ability to accurately simulate wave propagation in a long wave guide as in section 6.2.1, without loss of generality, the length of the two bars can be significantly reduced in this simplified SHPB model to shorten the simulation time of the analysis. Please note that compared to the uniform refined mesh, the computational efficiency of the s-adaptive method in the simulation of a real SHPB device is going to be higher than the efficiency achieved in this simplified model. The damage surface is defined as $F(\mathbf{Y},B) = (1/2)[J_{11}(Y_1)^2 + J_{22}(Y_2)^2] - (B_0 + B)$. Linear damage hardening are $B_0 = 0$, $B = c_1\beta = 0.001\beta$. For the specimen, the damage resistance constants are $J_{11} = J_{22} = 6$; and for the incident bar and the transmitter bar, the damage resistance constants $J_{11} = J_{22} = 0$, which means no damage will be accumulated in the two bars and therefore the two bars will be always be linear elastic. Because the Young's module of the specimen is only half of the ones of the incident bar and the transmitter bar, the reflections and the transmissions of the waves will occur at the interfaces of different parts.

The problem is solved as a 2-D, undamped, nonlinear plane stress problem caused by the progressive damage. The s-adaptive solution is initiated from a uniform global mesh that contains 10 elements in the x direction and 2 elements in the y direction. Adaptivity of the structured mesh superposition solution is controlled by the following parameters: $\eta_t^{TOL} = 0.05\%$, $\beta_t = 0.6$, $\theta_t = 0.9$, $K_t = 8$, $\eta_s^{TOL} = 5\%$, $\beta_s = 0.6$, $\phi = 0.6$, $\theta_s = 0.5$, and $K_s = 8$. The overlay mesh parameters are chosen as $d_h = 3$ and $d_v = 1$. The adaptive solution is computed for times $0 \le t \le 4$ sec., which is sufficient for the stress wave to travel through the transmitter bar/specimen interface completely.

Figure 7.29 shows a series of composite meshes and associated axial stress distributions that were produced at various points in time during the s-adaptive analysis. As seen in Figure 7.29, the overlay mesh is repeatedly updated to provide a composite mesh with high refinement in the immediate vicinity of the traveling stress wave, while maintaining a coarse mesh elsewhere. In order to maintain the user-specified spatial error tolerance, two levels of overlay meshes were needed, resulting in the smallest elements having a length of L/90 in the direction of wave propagation.

Figure 7.29: Series of composite meshes and computed axial stress distributions produced by the structured, s-adaptive solution at various points in time

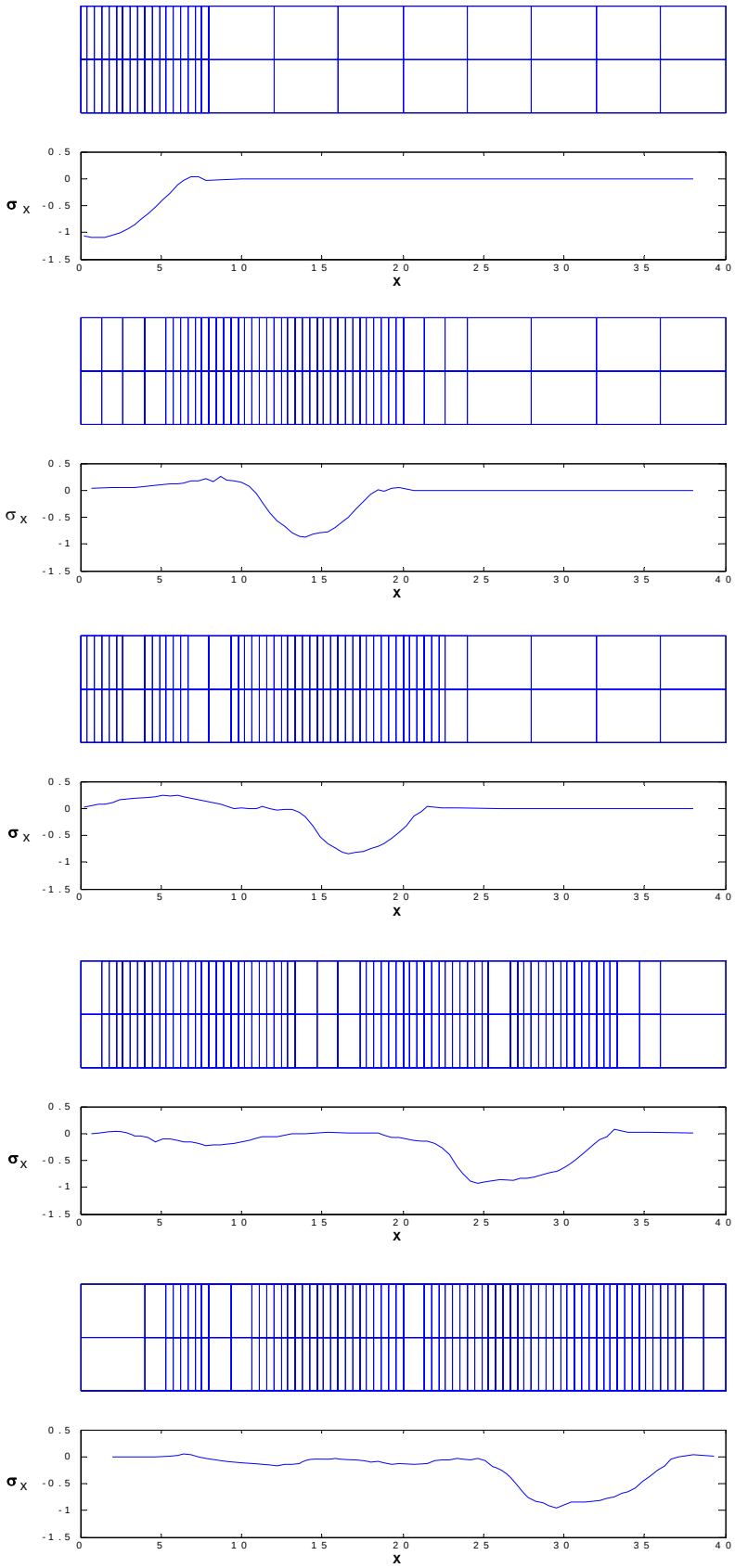

a)
t=0.603
b)
t=1.958
c)
t=2.358
d)
t=3.555
e)
t=3.950
σx
x

Figure 7.30 shows the number of active degrees of freedom used in the composite mesh as a function of time.

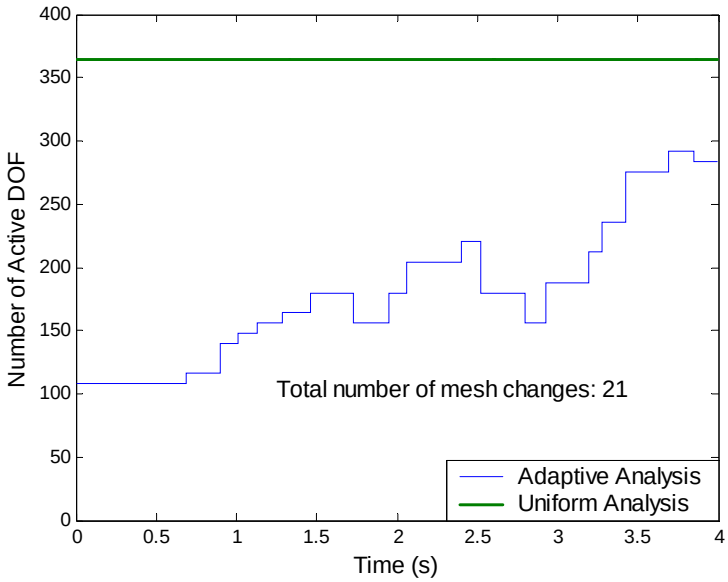


Figure 7.30: Time history of the number of active DOF used in the s-adaptive analysis

Usually, after the wave reaches the interfaces of two parts and gets reflected and transmitted, the number of active degrees of freedom starts to increase because more refinement is needed to simultaneously cover the reflected wave and the transmitted wave which are traveling in the opposite directions. This is true at t=3.1 sec. when the wave reaches the interface between the transmitter bar and the specimen. However, at t=1.7 sec. when the wave reaches the interface between the incident bar and the specimen, it is not the case. The reason is that the smaller stiffness of the specimen than the incident bar results in lower traveling speed in the specimen. At the beginning of the wave reflection/transmission, the backward reflected tensile wave is still completely contained

in the remainder of the forward compressive wave but the transmitted wave into the specimen is not as long as it was in the much stronger incident bar. As a result, the total length of the device covered by the stress wave is indeed decreasing. Therefore, at the beginning of the wave reflection/transmission at interface between incident bar and specimen, the number of DOF in the mesh is decreasing. After the backward reflected wave travels out of the remainder of forward compressive wave, the length covered by the stress wave begins to increase and consequently the number of DOF in the mesh is increasing, which happens approximately at t=2 sec. Also note that approximately at t=2.5 sec., the number of DOF in the mesh is decreasing. It is caused by the fact that the backward reflected tensile stress wave starts to reach the free end at the left and reflects as a forward compressive wave again. The superposed stress wave near the free end at the left can be adequately simulated using less refinement as pointed out in section 7.3.3. Also note that in response to the estimated spatial error presented in Figure 7.31, the composite mesh was updated 21 times during the course of the analysis; however, each update involves only minor changes to the overlay mesh and can be performed quickly.

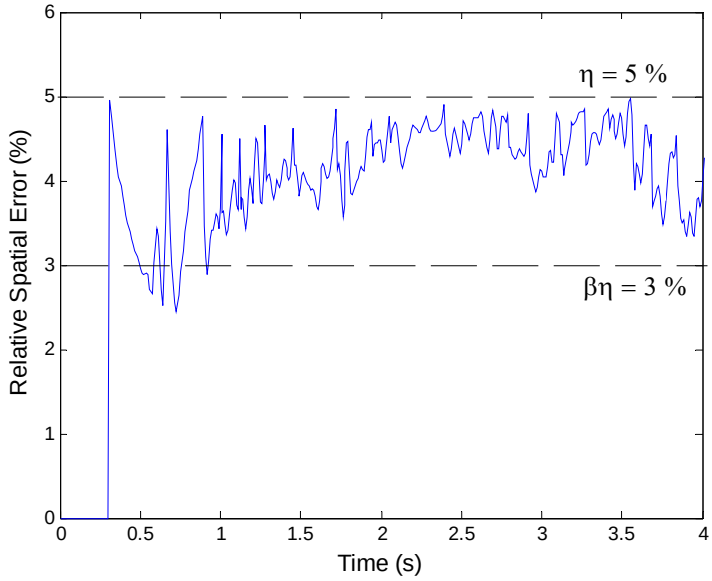


Figure 7.31: Time history of relative spatial error achieved in the adaptive analysis

The estimated temporal error is used to control the time step size over the course of the adaptive analysis. Figure 7.32 shows the time history of the time step size, while Figure 7.33 shows the estimated temporal error.

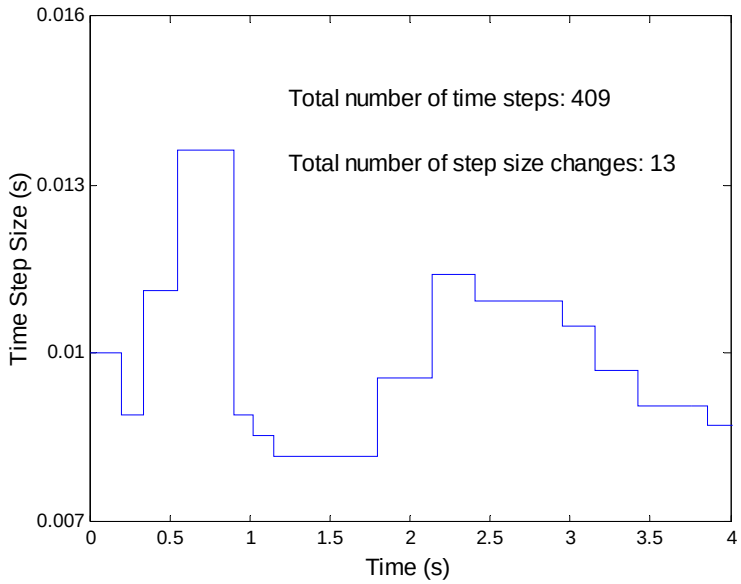


Figure 7.32: Time history of the time step size used in the adaptive analysis

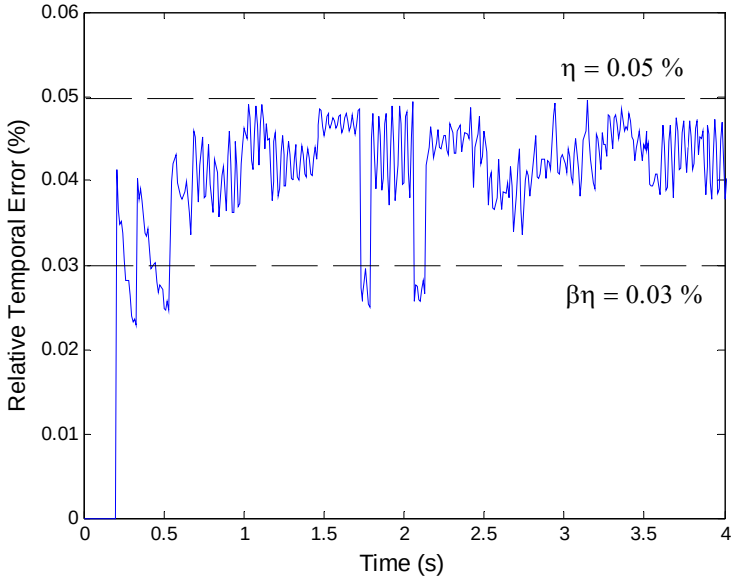
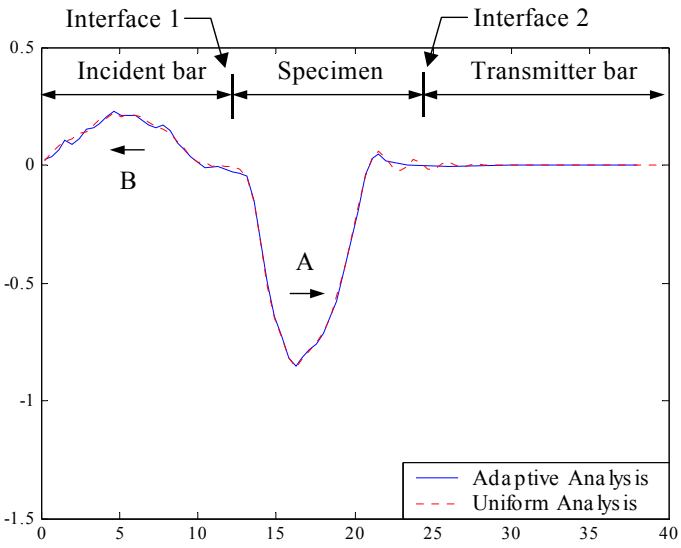


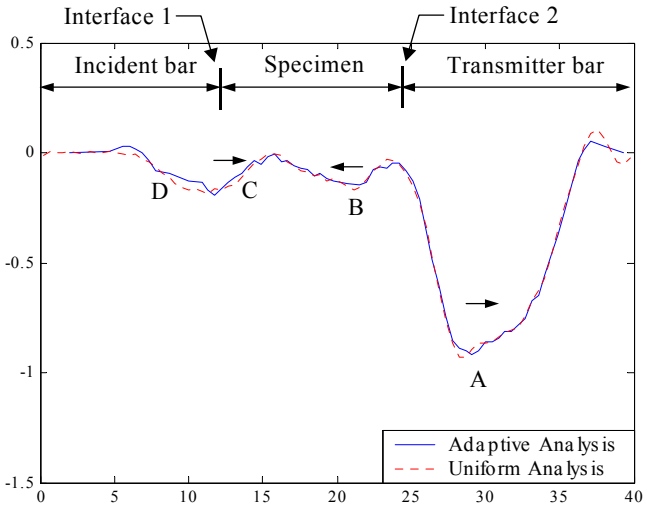
Figure 7.33: Time history of relative temporal error achieved in the adaptive analysis

For comparison, the same problem is solved with a uniform refined mesh of 90 elements in the x direction and 2 elements in the y direction (364 active DOF) with a time step size dt=0.01 seconds. The uniform element size is comparable to the most refined overlay mesh level used previously in the s-adaptive solution. Figures 7.34 and 7.35 show the location, size and shape of the stress wave predicted by both methods at the discrete times t=2.3, 3.9 seconds. The two solutions show excellent agreement. Three compressive wave peaks are clearly observed in Figure 7.35 and they are the wave reflected off the free edge of the left end, the reflected wave off the specimen/transmitter bar interface, and the transmitted wave through the specimen/transmitter bar interface, respectively.



A : main pulse after passing through interface 1
B : pulse reflected off interface 1

Figure 7.34: Comparison of the axial stress wave predicted at t=2.3 sec. by the s-adaptive solution and the uniform refined solution



A : main pulse after passing through interface 2
B : pulse reflected off interface 2
C : pulse transmitted through interface 1
D : superposed pulse of the pulse reflected off interface 1
and the pulse reflected off the left end of the incident bar

Figure 7.35: Comparison of the axial stress wave predicted at discrete times t= 3.9 sec. by the s-adaptive solution and the uniform refined solution

Figure 7.36 shows the time history of the total energy (which includes the strain energy and the kinetic energy) norm of the bar. The total energy norm increases as the half sine pulse is applied; and then it begins to decrease as the stress wave propagates through the incident bar/specimen interface and into the specimen at t=1.4 sec. The total energy norm stops decreasing when the stress wave propagates through the transmitter bar/specimen interface and into the transmitter bar at t=3.47 sec. The observation is physically sound because the damage can accumulate only in the specimen. Figure 7.37 shows the comparison of the damage distribution in the device by the s-adaptive solution and the uniform refined solution at t=4 sec. Both solutions predict no damage accumulation in the incident bar and the transmitter bar. In the specimen, though some noises in the adaptive solution causes the adaptive solution to oscillate above and below the uniform refined solution, the two solutions actually show good agreement. The explanation for the noise has been presented in section 7.3.3. Figure 7.38 shows the comparison of the material stiffness distribution in the specimen by the s-adaptive solution and the uniform refined solution at t=4 sec., which are also in good agreement.

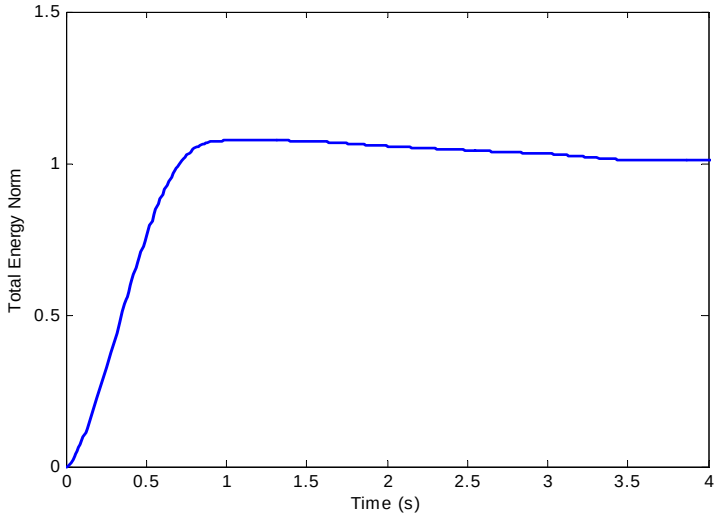


Figure 7.36: Time history of the total energy norm of the bar

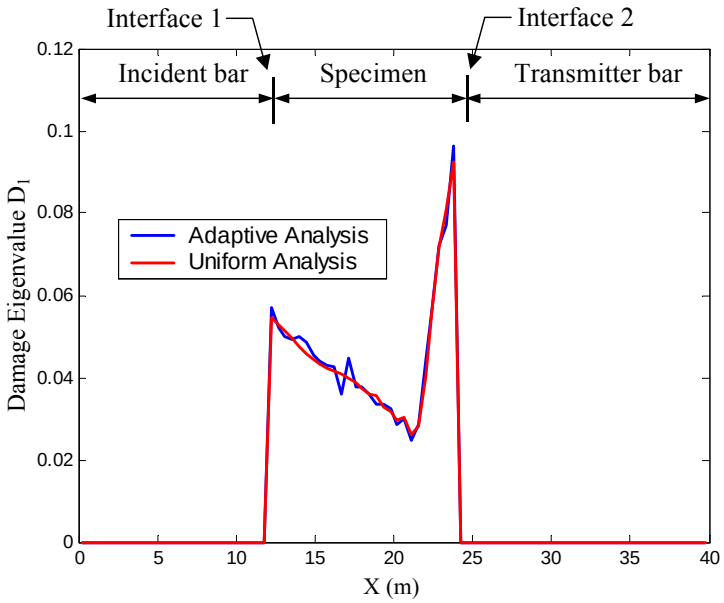


Figure 7.37: Comparison of the damage distribution in the bar by the s-adaptive solution and the uniform refined solution at t=4 sec.

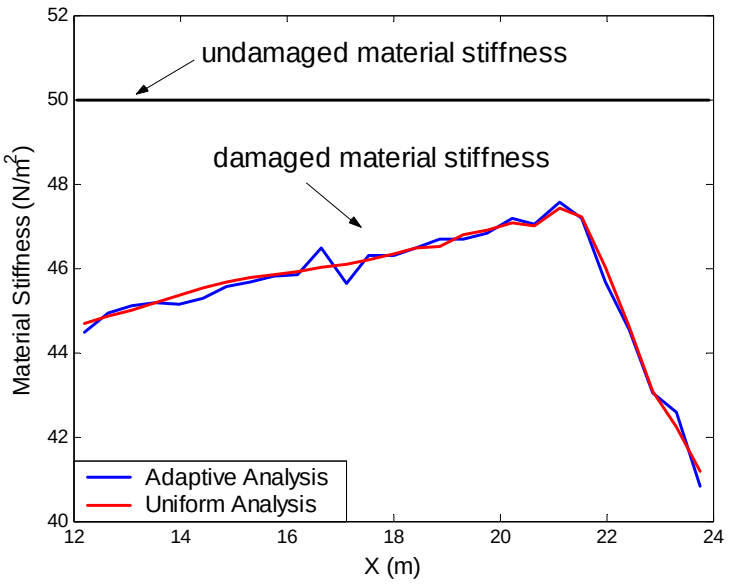


Figure 7.38: Comparison of the material stiffness distribution in the specimen by the s-adaptive solution and the uniform refined solution at t=4 sec.

7.3.5 Stress Wave Propagation in a 2-D, Damped, Bending Beam

This example involves the propagation and reflection of stress waves in a 2-D, unconstrained beam subjected to time varying surface tractions under the assumption of 2-D plane stress. The beam exhibits material nonlinearity caused by progressive damage and is included here to demonstrate the effectiveness of the adaptive methodology on nonlinear transient problems where the solution varies with respect to both spatial coordinates in addition to time. Since this particular problem requires spatial adaptivity in both coordinate directions (in a nonlinear transient setting), it poses a relatively severe test of the adaptive methodology. A quasi-static, adaptive solution of this problem was presented in [56] where material nonlinearity was caused by plasticity.

Figure 7.39 shows the geometry, boundary conditions, material properties and external loading. The externally applied surface tractions $\vec{F}_0$ on the left end and $\vec{F}_L$ on the right end are given in equations (7.26a,b). The factor $p(t)$ in equations (7.26a,b) causes $\vec{F}_0$ and $\vec{F}_L$ to vary with respect to time in the form of a half sine-wave pulse as shown in Figure 7.39. The initial undamaged isotropic material properties of the beam are E = 100, $\nu = 0.3$, $\rho = 1$. Rayleigh damping is assumed as $\boldsymbol{C} = 0.001\boldsymbol{M} + 0.001\boldsymbol{K}$. The damage surface for the beam material is defined as $F(\mathbf{Y},B) = (1/2)[J_{11}(Y_1)^2 + J_{22}(Y_2)^2] - (B_0 + B)$, where $J_{11} = J_{22} = 1$ and the linear damage hardening constants are $B_0 = 0$, $B = c_1\beta = 0.001\beta$.

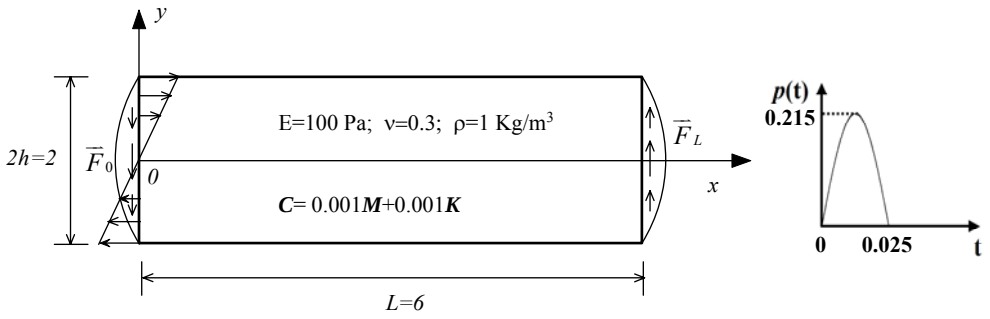


Figure 7.39: A 2-D beam bended under half-sine-wave pulses

The applied surface tractions are expressed as follows:

$$\vec{F}_0 = p(t)\left[2Ly\vec{x} + \left(y^2 - h^2\right)\vec{y}\right] \quad \text{on} \quad x = 0 \tag{7.26a}$$

$$\vec{F}_L = p(t)\left[\left(h^2 - y^2\right)\vec{y}\right] \quad \text{on} \quad x = L \tag{7.26b}$$

$$p(t) = \begin{cases} \sin\left(\dfrac{t}{0.025}\pi\right); & t \in [0, 0.025] \\ 0; & t \in (0.025, \infty) \end{cases} \tag{7.26c}$$

The problem is solved as a 2-D, damped, nonlinear plane stress problem caused by the progressive damage. For the s-adaptive solution, a uniform global mesh is chosen with 18 elements in the x direction and 6 elements in the y direction. Relative error tolerances and corresponding adaptivity parameters are specified as: $\eta_t^{TOL} = 0.5\%$, β_t = 0.6, θ_t = 0.9, K_t =8 and $\eta_s^{TOL} = 10\%$, β_s = 0.6 , ϕ = 0.6, θ_s = 0.9, K_s =8 respectively. The overlay mesh density parameters are chosen as d_h= d_v= 2. The adaptive solution is conducted over a time interval $0 \le t \le 1.5$ sec., which is sufficient for the stress waves to travel the length of the beam, reflect off of the free edges, and travel back to the beam's mid-span. Note that the duration of the computed solution is sixty times the duration of the externally applied impulsive forces.

As shown in Figure 7.39, the externally applied normal traction $\sigma_{xx}(0,y)$ varies linearly with respect to the y direction, while the externally applied shear tractions $\sigma_{xy}(0,y)$ and $\sigma_{xy}(L,y)$ vary quadratically with respect to the y direction. The antisymmetric normal stress loading and the symmetric shear stress loading about the neutral axis y=0 enable the beam to present the corresponding antisymmetric and symmetric stress distributions about the neutral axis too. Figure 7.40 shows the stress distributions of σ_{xx}, σ_{yy}, σ_{xy} along various lines at y=h, 2h/3, h/3, 0, -h/3, -2h/3, –h at time t=0.0378 sec. It is clearly observed that the normal stresses σ_{xx}, σ_{yy} present antisymmetry about the neutral axis and the shear stress σ_{xy} presents symmetry about the neutral axis. These observed properties of symmetry and antisymmetry of stress distributions are conserved in the subsequent analysis and are also

confirmed in the numerical results. Therefore, it is sufficient to show the stress distributions in the top part of the beam ($y>0$) in the following study to save space.

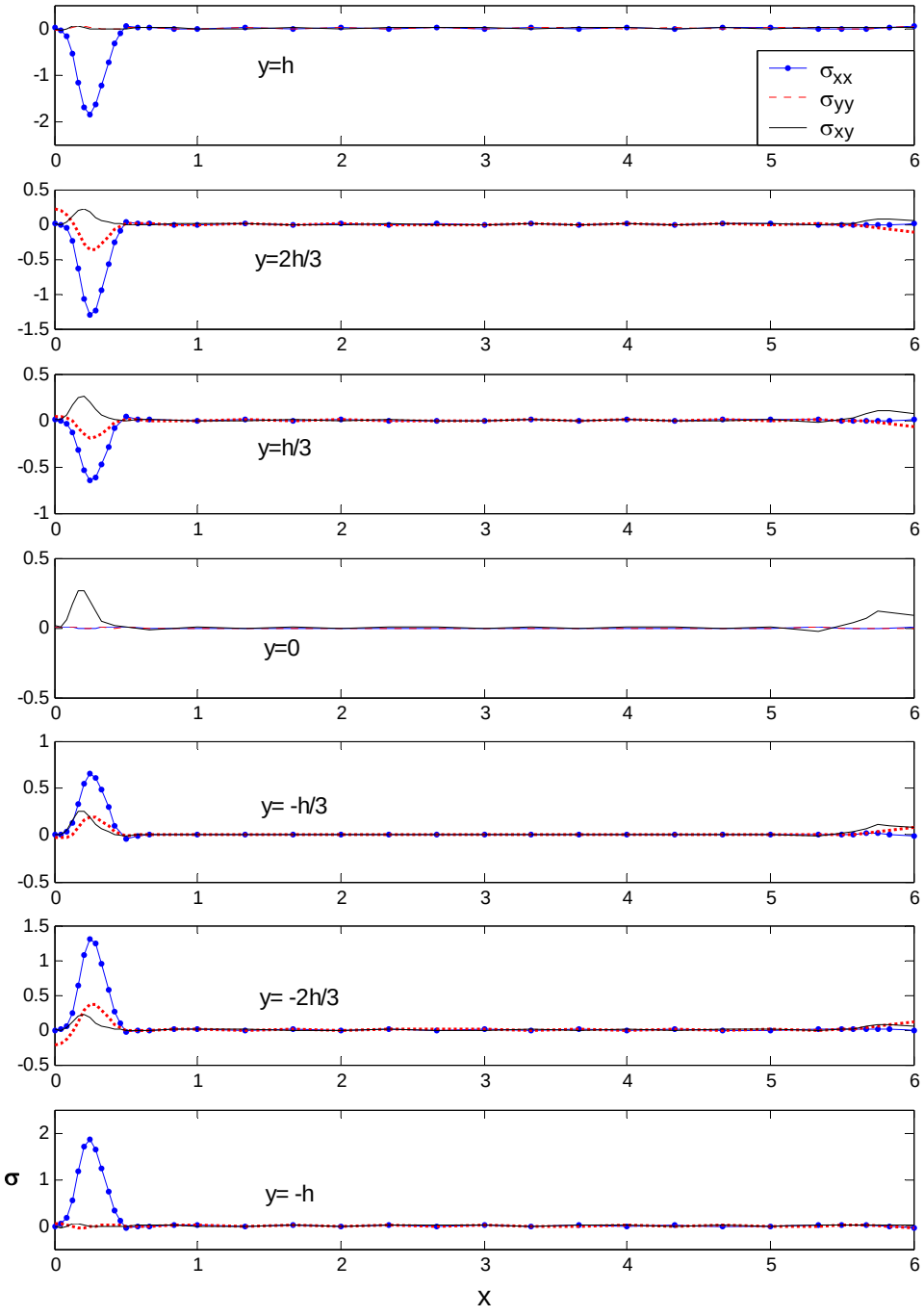


Figure 7.40: Computed stress distributions produced by the s-adaptive solution along various lines at y=h, 2h/3, h/3, 0, -h/3, -2h/3, -h at time t=0.0378 sec.

Figure 7.41 through Figure 7.52 show a series of composite meshes and associated stress distributions that were produced at various points in time during the s-adaptive analysis. As seen in these figures, the overlay mesh is repeatedly updated to provide a composite mesh with high refinement in the immediate vicinity of high stress gradients, while maintaining a coarse mesh elsewhere. In order to maintain the user-specified spatial error tolerance, the composite mesh typically consisted of the global mesh plus three levels of overlay meshes that were distributed as needed, resulting in the smallest elements of the composite mesh having a length of L/144 in the x direction and a height of h/24 in the y direction.

As shown in the stress distributions along line y=h in Figures 7.41 through Figure 7.52, the only significant stress component near the upper surface of the beam is the normal stress σ_{xx}. This is physically reasonable since the top boundary of the beam is a traction-free boundary where σ_{yy} and σ_{xy} must vanish. Along the neutral axis y=0, the only significant stress component is the shear stress σ_{xy}. This is physically reasonable since the applied normal traction σ_{xx} is zero at y=0 while the applied shear traction is maximum at y=0.

Despite the fact that similar external shear tractions are applied to both ends of the beam (x=0 and x=L), Figures 7.41 through 7.52 clearly show that the shear stress σ_{xy} distributions near the two ends of the beam are qualitatively and quantitatively different

from one another. Specifically, Figure 7.41 shows that, immediately after the externally applied pulses terminate, the magnitude of the shear stress distribution near x=0 is much larger than the one near x=L. Furthermore, as the stress waves continue to propagate toward the mid-span (x=L/2) of the beam, Figures 7.43 through 7.46 clearly show that the form (i.e. shape) of the shear stress wave is much more complex in the left-hand half of the beam than in the right-hand half of the beam. These observed differences in the form and magnitude of the shear stress waves in the left-hand and right-hand halves of the beam are caused by the presence of the normal stress wave which emanates from the left-hand boundary (x=0) and travels to the right. At any point in time, the normal stress varies linearly in the y direction, with maximum compression at the top of the beam and maximum tension at the bottom of the beam; consequently, the normal stress wave continually creates an associated shear stress which adds to the shear stress wave that emanates from the left-hand boundary caused by the externally applied shear traction.

In order to see that the normal stress wave generates an accompanying shear stress wave, consider a small volume element ($dV \rightarrow dx\,dy$) in the top half of the beam ($y>0$) at the current location x of the normal stress wave. Due to the linear variation of σ_{xx} with respect to y, the material fibers along the top edge of the volume element experience a degree of compression that is different from the material fibers along the bottom edge of the volume element. Thus the originally rectangular volume element is deformed into a trapezoidal shape which necessarily requires the presence of shear deformation to obtain

(hence the additional shear stress). This effect approximately uniformly occurs in the beam and vanishes as it approaches the top and bottom surfaces of the beam since these are traction-free boundaries. Therefore, we conclude that the elevated levels of shear stress seen in the left-hand portion of the beam near the neutral axis are the result of the linearly varying normal stress wave.

An examination of Figures 7.41 through 7.52 reveals that the amplitude of the normal stress waves σ_{xx}, σ_{yy} and the amplitude of the shear stress wave σ_{xy} decrease as time progresses. This amplitude decrease is the result of energy dissipation caused by both damping and progressive damage. In addition, as time progresses, the wavelength of the various stress waves σ_{xx}, σ_{yy}, σ_{xy} increases due to dispersion which is again caused by the presence of damping and progressive damage. Eventually, energy dissipation and dispersion result in low-energy wave forms nearly covering the entire computational domain as seen in Figure 7.52 at time t=1.4927.

Shear stress waves propagate at a lower wave speed than the normal stress waves. Therefore, the shear stresses that are produced in the wake of the faster moving normal stress wave lead to lag behind the normal stress wave, resulting in the interesting patterns seen in the composite meshes of Figures 7.44 through 7.48.

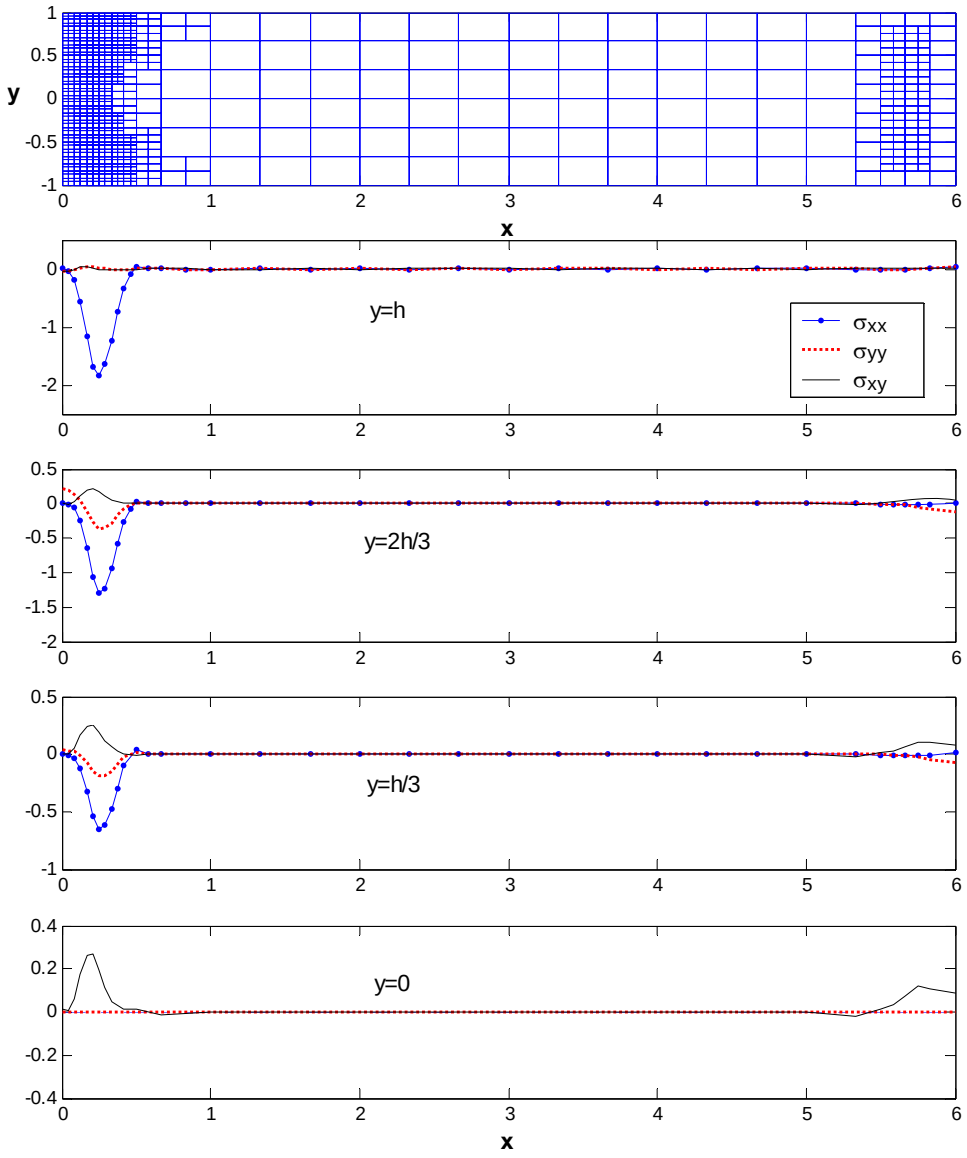


Figure 7.41: Composite mesh and computed stress distributions produced by the s-adaptive solution along various lines at y=h, 2h/3, h/3, 0 at time t=0.0378 sec.

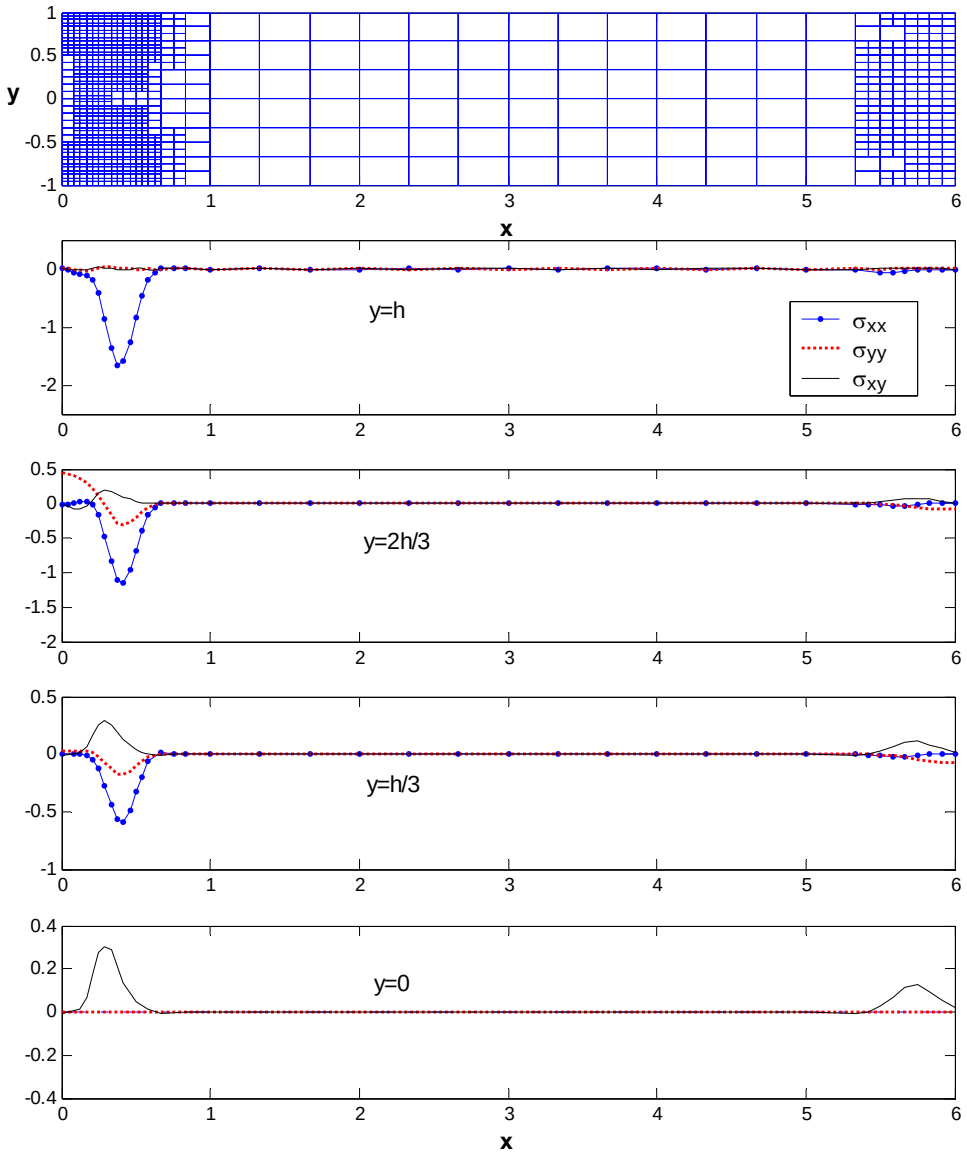


Figure 7.42: Composite mesh and computed stress distributions produced by the s-adaptive solution along various lines at y=h, 2h/3, h/3, 0 at time t=0.0509 sec.

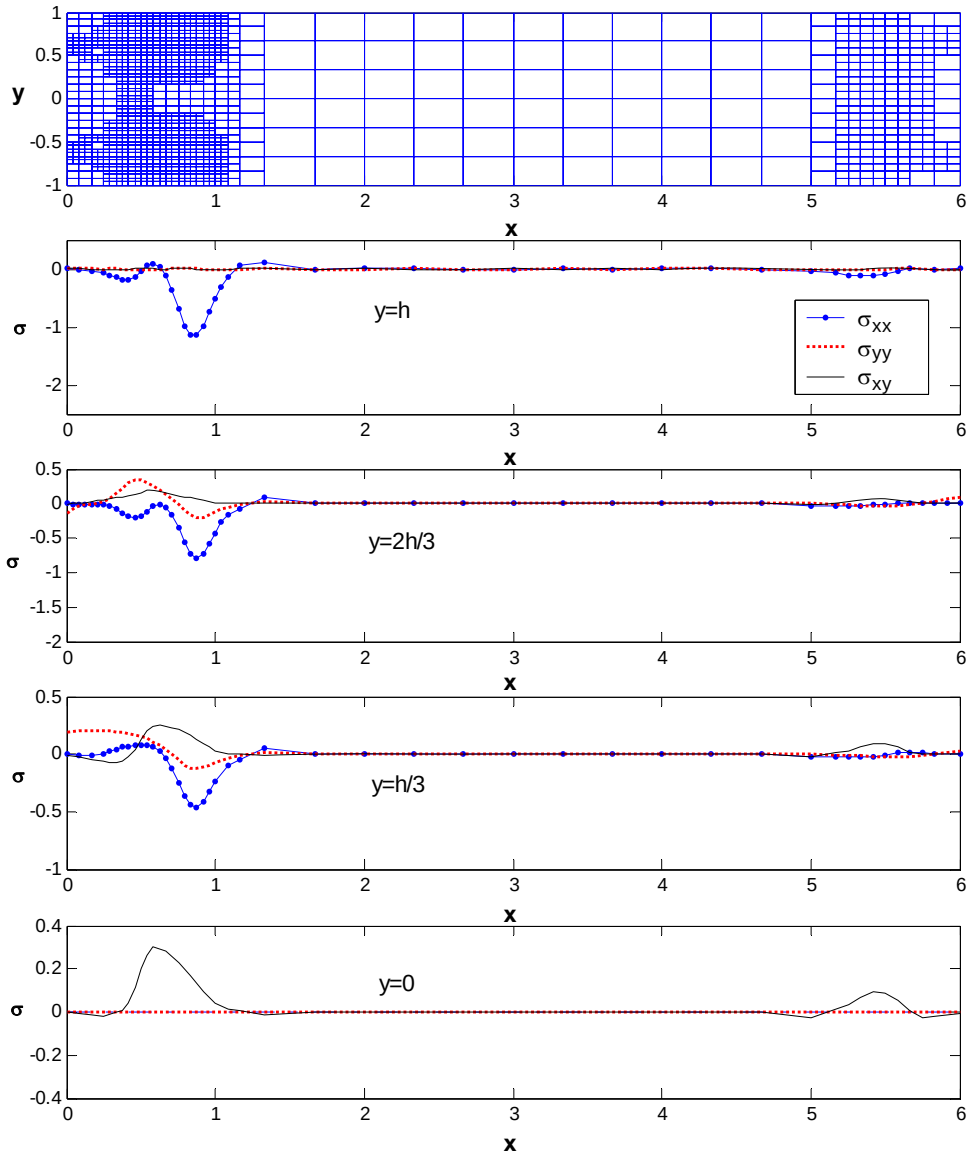


Figure 7.43: Composite mesh and computed stress distributions produced by the s-adaptive solution along various lines at y=h, 2h/3, h/3, 0 at time t=0.0945 sec.

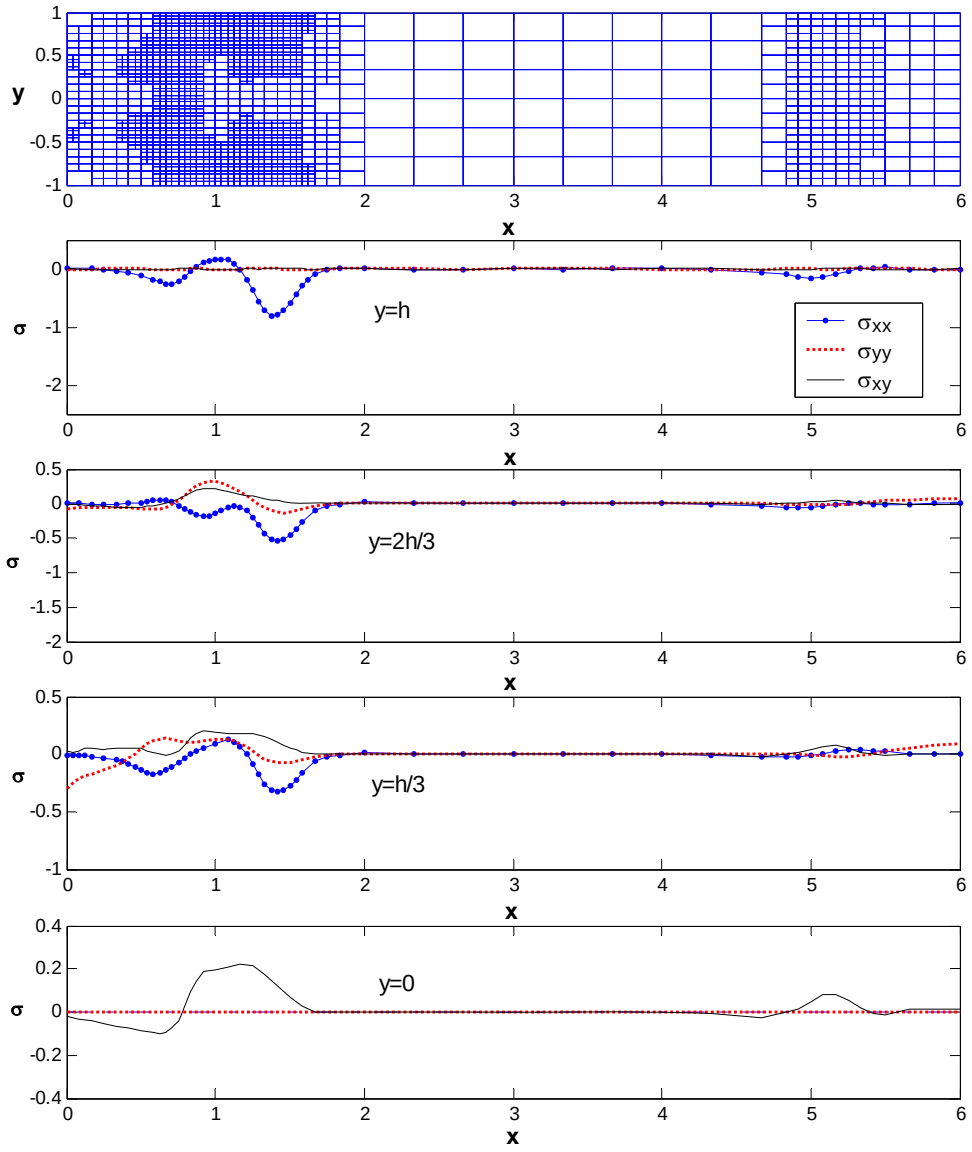


Figure 7.44: Composite mesh and computed stress distributions produced by the s-adaptive solution along various lines at y=h, 2h/3, h/3, 0 at time t=0.1451 sec.

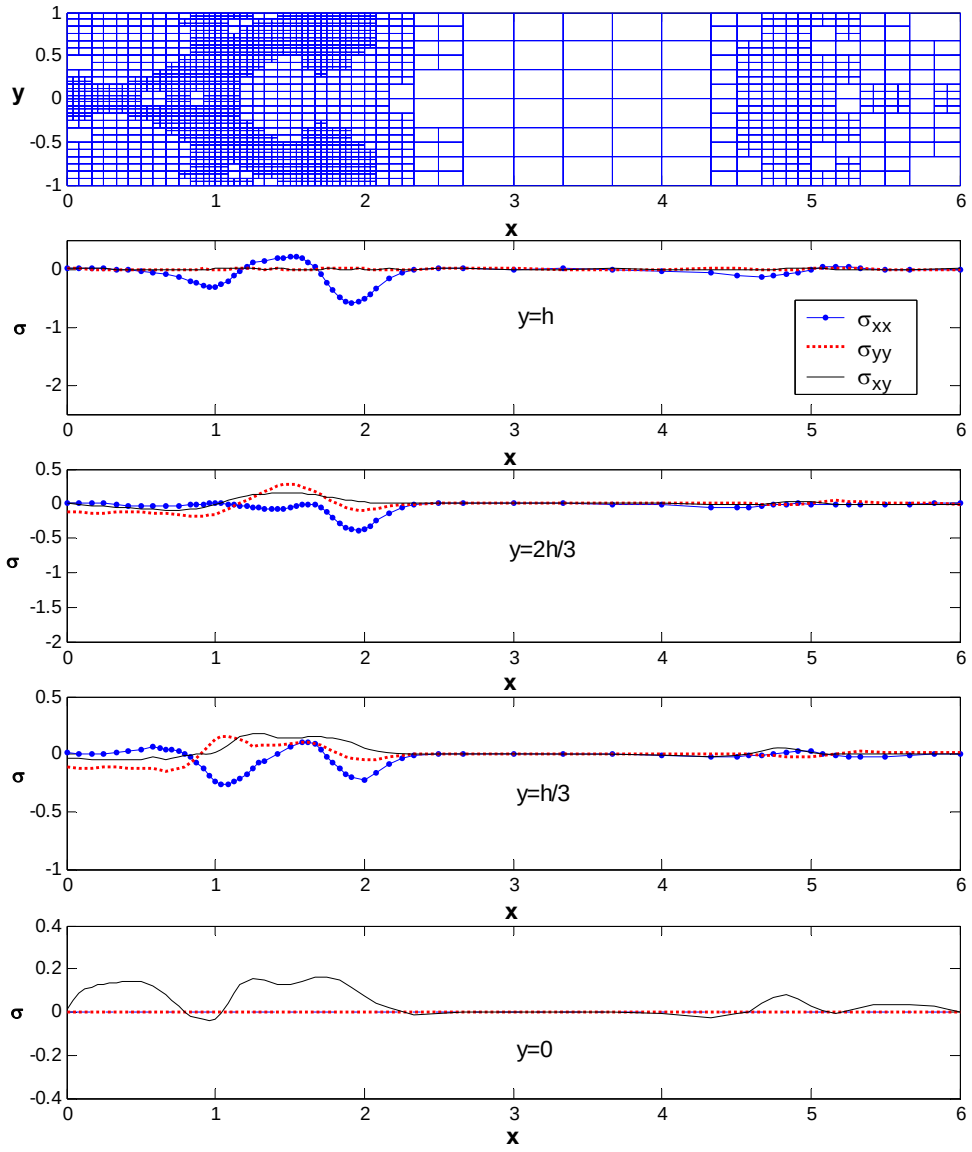


Figure 7.45: Composite mesh and computed stress distributions produced by the s-adaptive solution along various lines at y=h, 2h/3, h/3, 0 at time t=0.1940 sec.

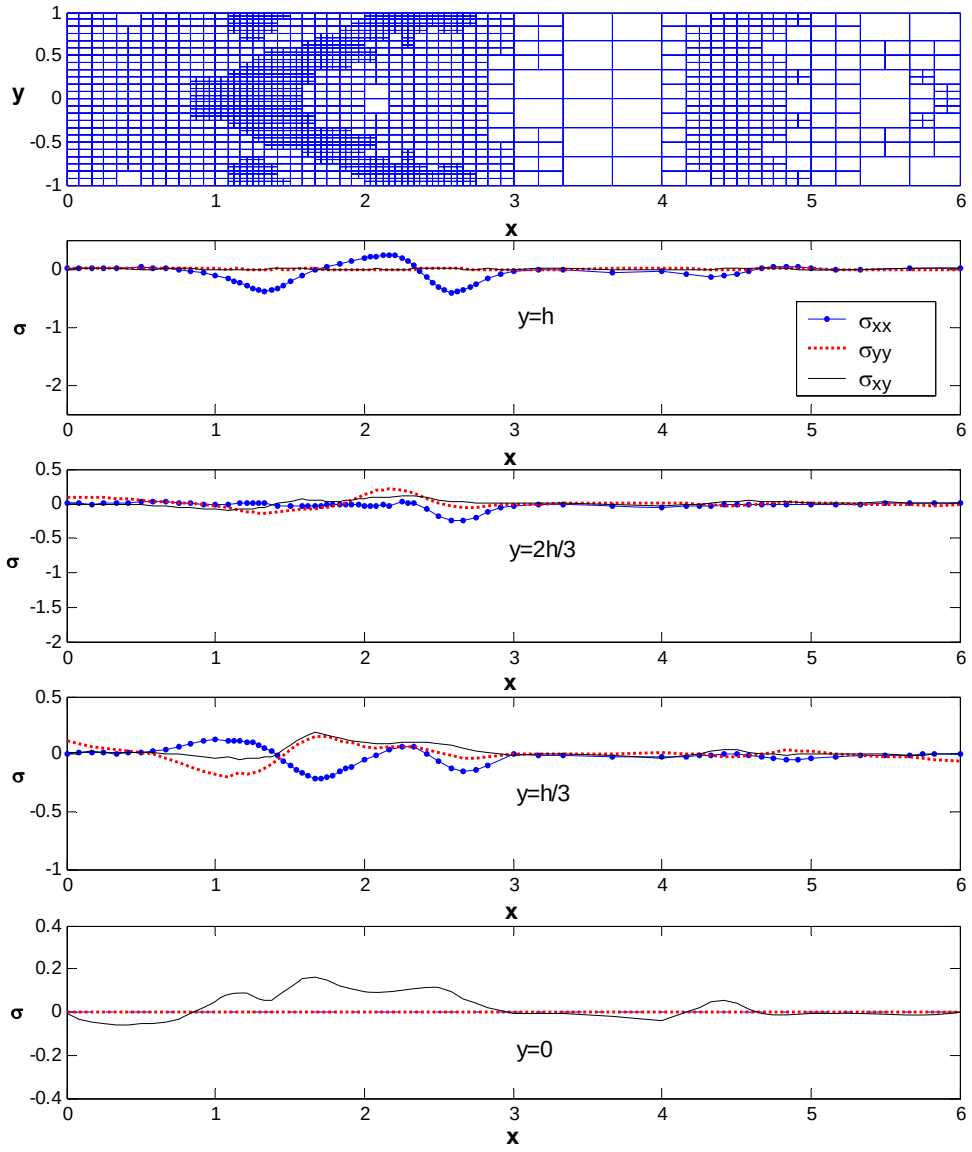


Figure 7.46: Composite mesh and computed stress distributions produced by the s-adaptive solution along various lines at y=h, 2h/3, h/3, 0 at time t=0.2567 sec.

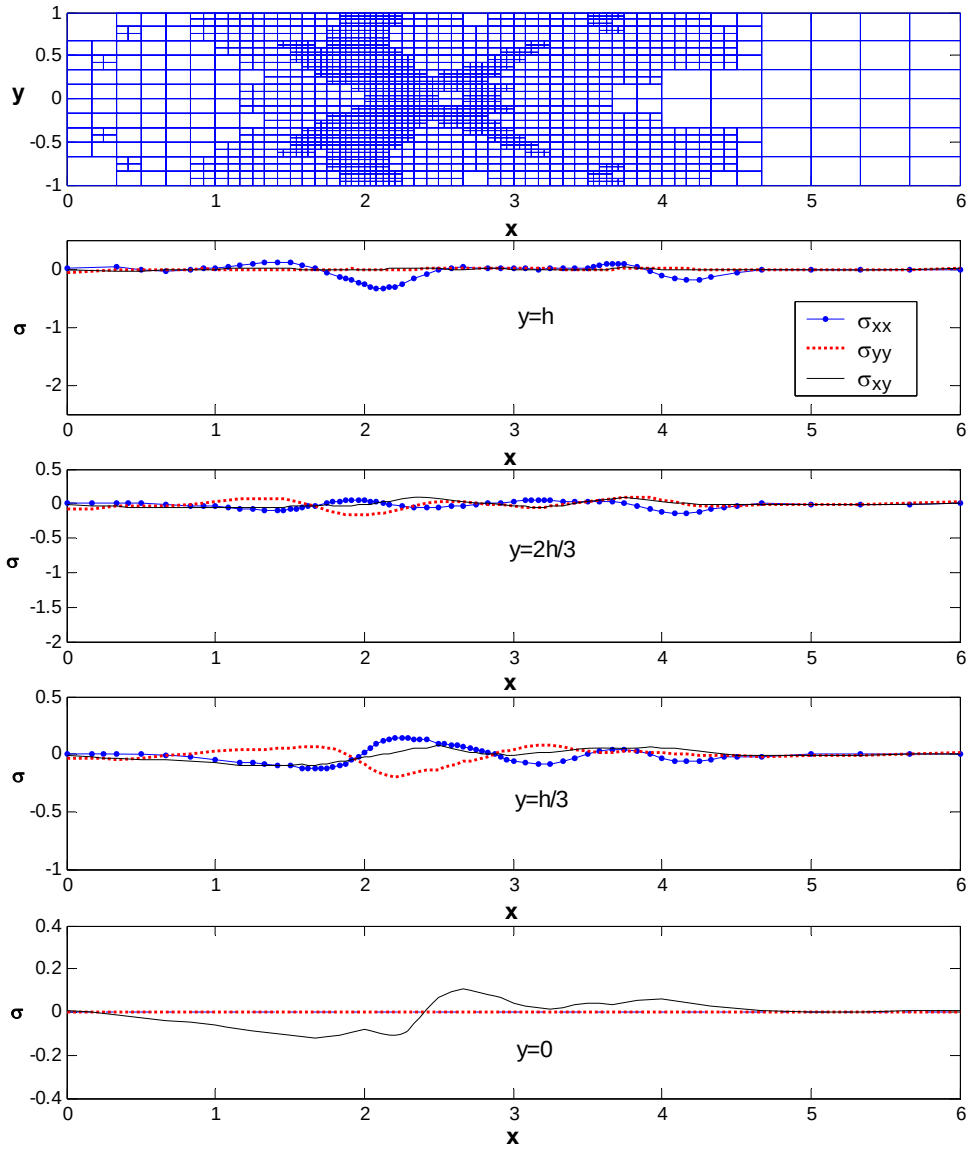


Figure 7.47: Composite mesh and computed stress distributions produced by the s-adaptive solution along various lines at y=h, 2h/3, h/3, 0 at time t=0.3942 sec.

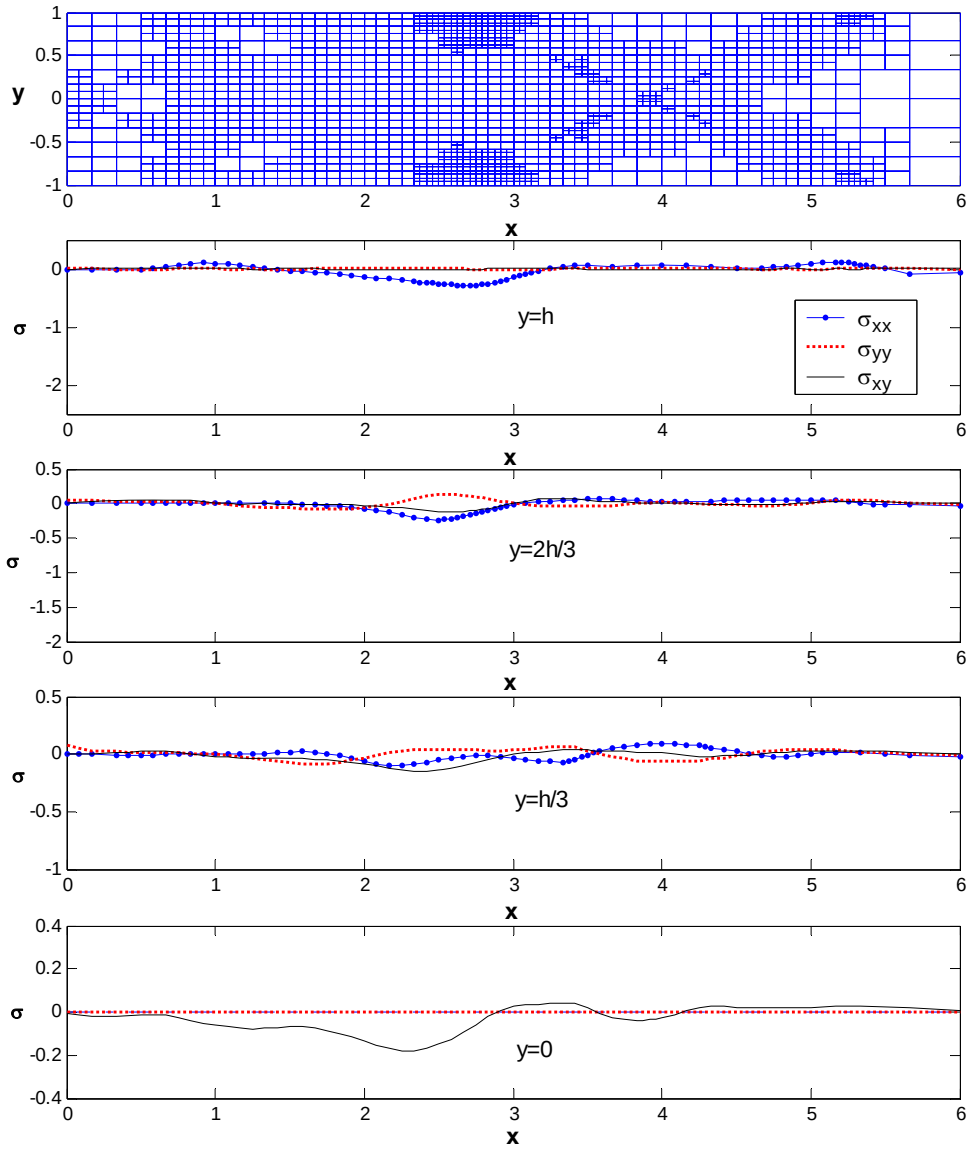


Figure 7.48: Composite mesh and computed stress distributions produced by the s-adaptive solution along various lines at y=h, 2h/3, h/3, 0 at time t=0.5376 sec.

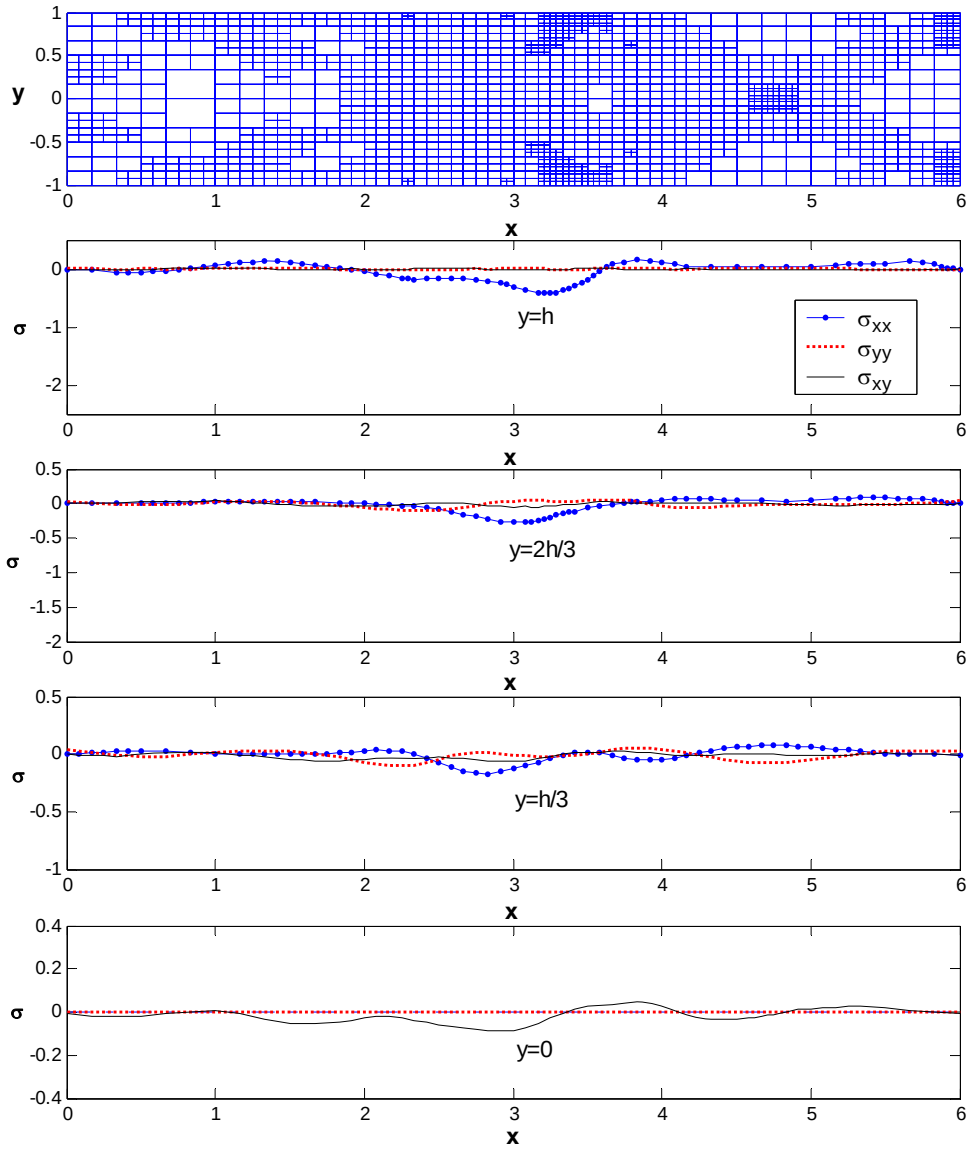


Figure 7.49: Composite mesh and computed stress distributions produced by the s-adaptive solution along various lines at y=h, 2h/3, h/3, 0 at time t=0.6010 sec.

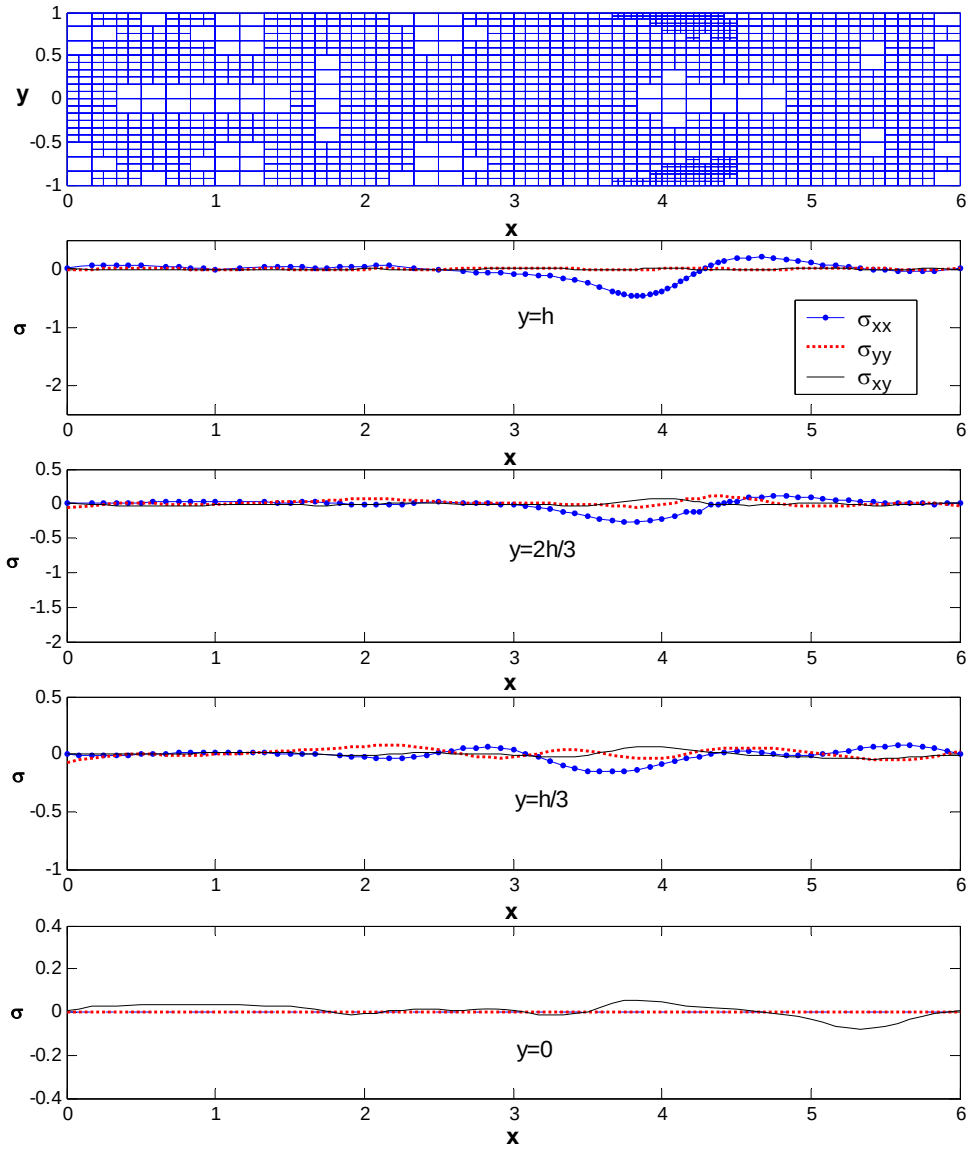


Figure 7.50: Composite mesh and computed stress distributions produced by the s-adaptive solution along various lines at y=h, 2h/3, h/3, 0 at time t=0.6826 sec.

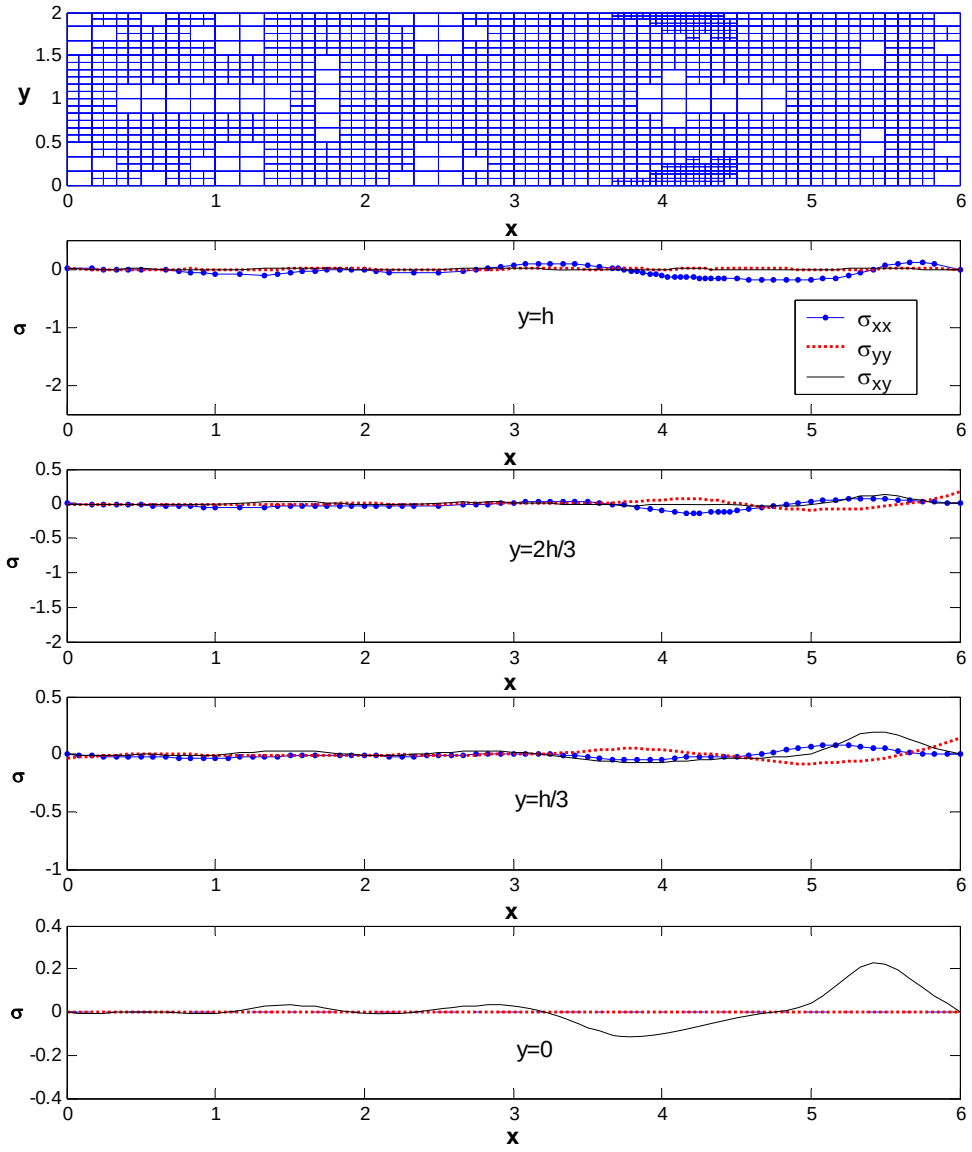


Figure 7.51: Composite mesh and computed stress distributions produced by the s-adaptive solution along various lines at y=h, 2h/3, h/3, 0 at time t=0.9442 sec.

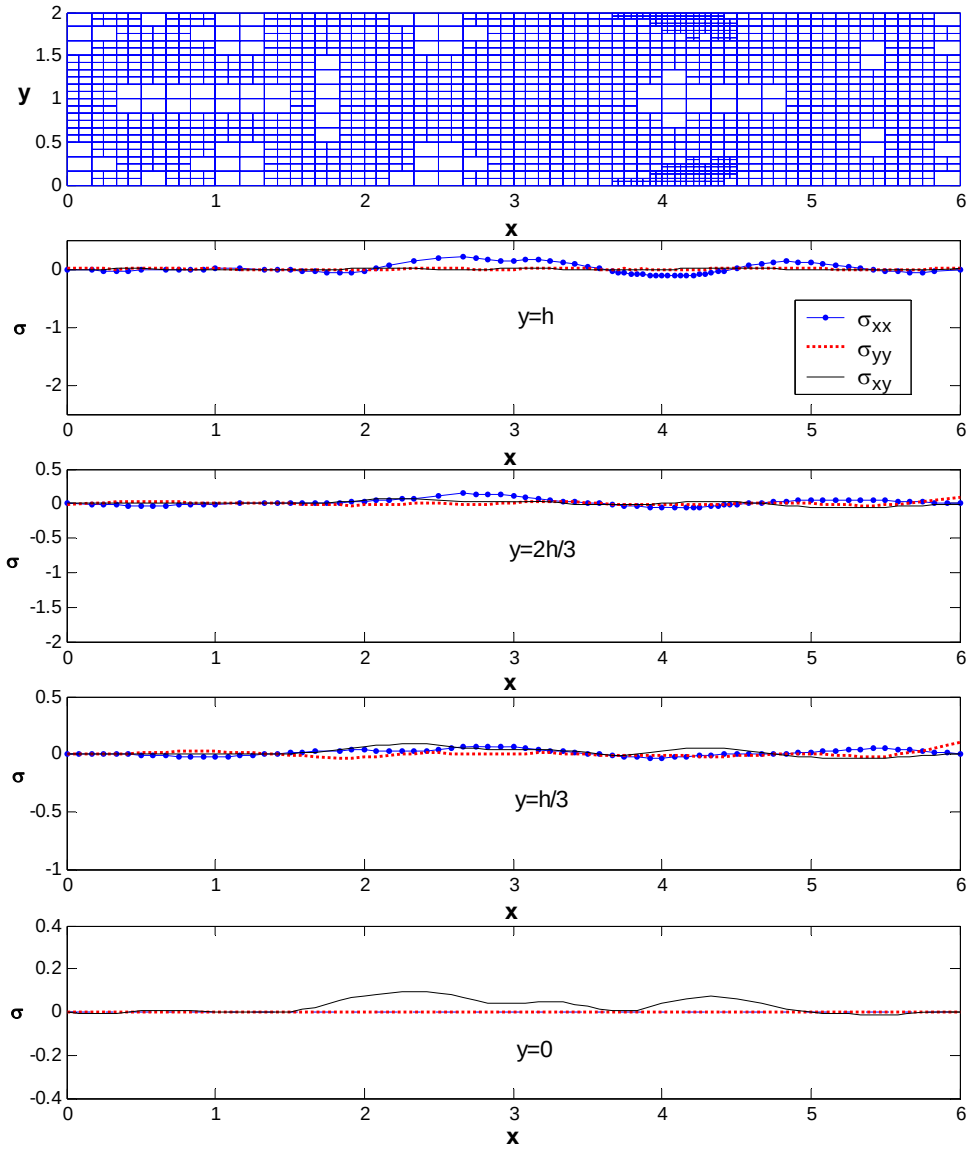


Figure 7.52: Composite mesh and computed stress distributions produced by the s-adaptive solution along various lines at y=h, 2h/3, h/3, 0 at time t=1.4927 sec.

Figure 7.53 shows the number of active DOF used in the composite mesh as a function of time. As seen in Figure 7.53, the composite mesh was updated a total of 22 times

during the course of the analysis; however, each of these updates involve only minor changes to the overlay meshes and thus can be performed very quickly. For discussion purposes, the total time period of the analysis can be divided into three distinct time periods: the initial time period ($0<t<0.2$), the intermediate time period ($0.2<t<0.66$) and the final time period ($0.66<t<1.5$). Examination of Figure 7.53 reveals a very rapid increase in the number of active DOF required during the initial time period ($0<t<0.20$). A review of Figures 7.41 through 7.44 shows that during this initial time period ($0<t<0.2$), the stress waves are introduced into the beam and quickly assume their fully-developed, high-energy, spatially-compact forms which are characterized by locally steep stress gradients, thus requiring a composite mesh with highly refined localized regions.

During the intermediate time period ($0.2<t<0.66$), as the mature wave forms propagate through the material, energy dissipation and dispersion (caused by damping and progressive damage) reduce the stress gradients exhibited by the individual wave forms, hence the slight overall decline in the number of active DOF observed during this time period in Figure 7.53. Note that this overall decline in the number of active DOF is not monotonic and contains several temporary increases in the number of active DOF caused by the waves colliding with one another near the beam's mid-span.

Interestingly, during the final time period ($0.66<t<1.5$), the number of active DOF remains unchanged, despite the fact that the waveforms are still in motion throughout the

final time period. This can be explained by the fact that the individual waveforms have degenerated into low-energy scattered waves which saturate the entire computational domain. Evidently, the composite mesh that was created at t=0.66 seconds proved to be adequate for the remainder of the analysis. In other words, as the low-energy scattered waveforms continue to move throughout the final time period, this single composite mesh was capable of delivering solutions that met the user-specified error tolerances, regardless of the precise position of the scattered waves.

The estimated relative spatial error is presented in Figure 7.54. Note that the initial and intermediate time periods (i.e., $0<t<0.66$) are characterized by rapid, non-smooth changes in the estimated spatial error which is caused by the repeated changes made to the composite mesh in keeping the spatial error under control. In contrast, the final time period ($0.66<t<1.5$) is characterized by a smoothing fluctuating spatial error that never exceeds the user-specified error tolerances and thus requires no further mesh updates. The smooth fluctuation of the estimated spatial error during the final time period is directly attributable to the fact that the entire final time period was solved with a single unchanging composite mesh, thus the fluctuation is due entirely to transient behavior and not mesh changes.

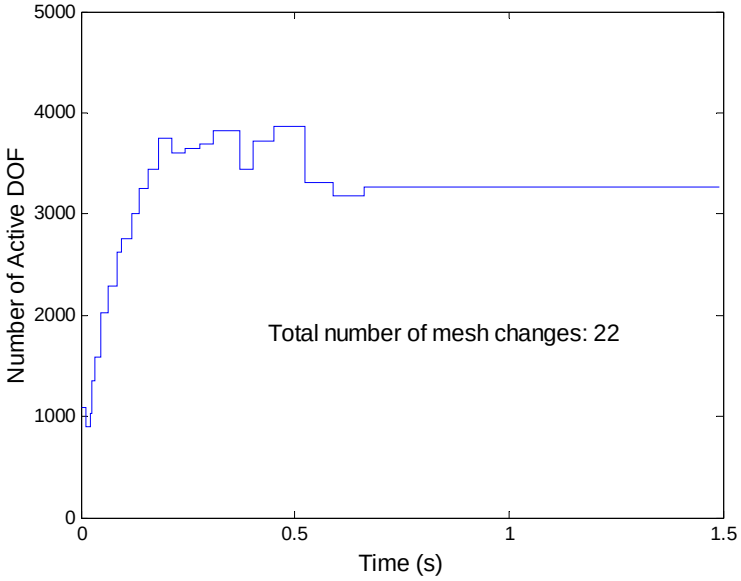


Figure 7.53: Time history of the number of active DOF used in the s-adaptive analysis

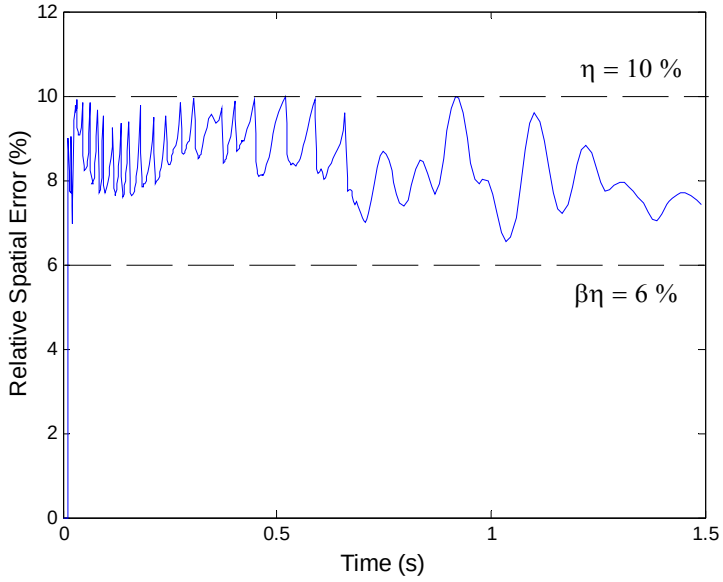


Figure 7.54: Time history of relative spatial error achieved in the adaptive analysis

The estimated temporal error is used to control the time step size over the course of the adaptive analysis. Figure 7.55 shows the time history of the time step size, while Figure

7.56 shows the estimated temporal error. Note that dramatic reduction of time step size occurs frequently prior to t=0.7 sec., forming many undesirable spikes in Figure 7.55. In this study, after a mesh change is performed, the solutions of the displacement and the velocity are transferred from the old mesh to the new mesh. And the solutions of the acceleration and the third order derivative of the displacement, which are needed in temporal error estimation, are directly calculated by resolving the dynamic equations on the new composite mesh. This causes a dramatic increase of the temporal error immediately after a mesh change is performed, and consequently, causes a dramatic decrease in the computed time step size. Most of the time, this dramatic decrease of time step size is unnecessary because the time step size increases gradually due to the temporal adaptivity in the subsequent time steps. Therefore, new methods of estimating the temporal error might prove useful in avoiding this issue. Note that after t>0.72 sec., the time step size increases gradually and no spikes are generated as shown in Figure 7.55. This is because the composite mesh remains unchanged after t>0.72 sec.

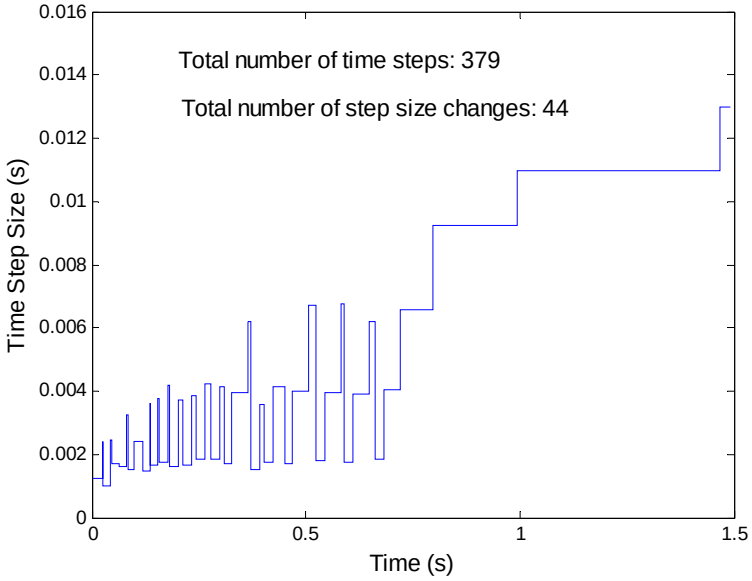


Figure 7.55: Time history of the time step size used in the adaptive analysis

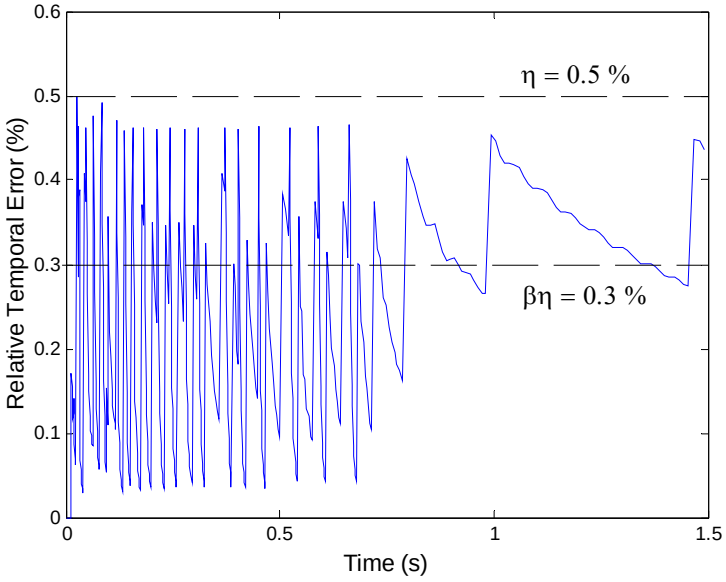


Figure 7.56: Time history of relative temporal error achieved in the adaptive analysis

Figure 7.57 shows the time history of the total energy norm of the beam (which includes both the strain energy and the kinetic energy). The total energy norm increases rapidly during the application of the externally applied impulse loads. But the presence of

damping and damage dissipates the total energy of the beam as in Figure 7.57. Again, the estimates for the relative error were computed based on the instantaneous total energy norm. This choice permits accuracy to be maintained even after the total energy has significantly diminished.

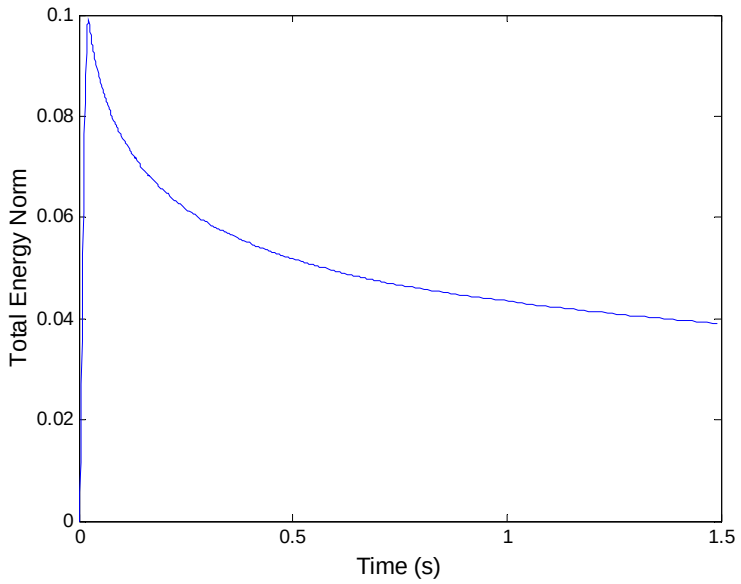


Figure 7.57: Time history of the total energy norm of the beam

As shown in numerical results, the predominant damage eigenvalue in this example problem is the damage eigenvalue D_1 which characterizes the distribution and density of microcracks that are normal to the x direction. The other two damage eigenvalues, D_2 and D_3 are negligible compared to D_1. Figure 7.58 shows the cumulative distributions of damage eigenvalue D_1 along the y direction on cross-sections at various x locations. The predicted damage distributions are symmetric about the neutral axis of the beam y=0 due to the symmetric loading of the beam's cross section. At any given cross-section,

the damage is highest near the upper and lower surfaces of the beam and is zero near the neutral axis (y=0). This is consistent with the symmetric, linear distribution of σ_{xx} which passes through zero at y=0. In addition, Figure 7.58 shows that the damage level is highest near the left-hand end of the beam (x=0) and decreases along the length of the beam (as x increases). This is consistent with the fact that both damping and damage dissipate the energy of the stress wave and reduce its capacity for causing further damage as the wave propagates.

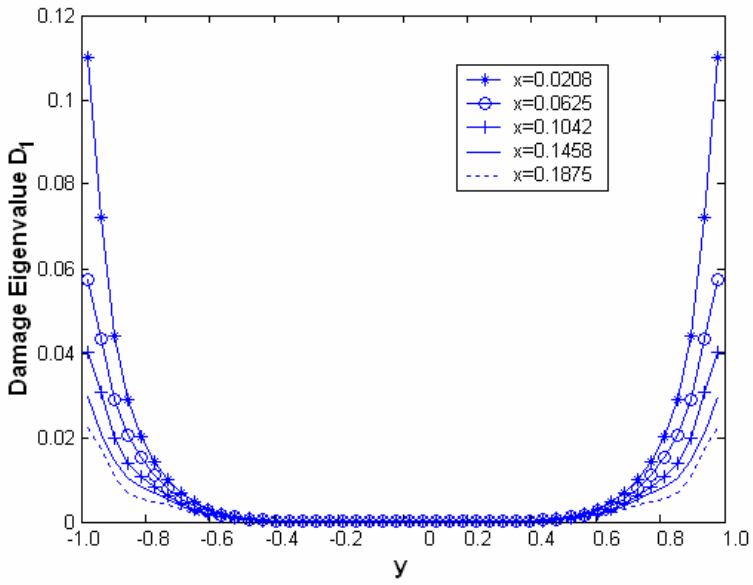


Figure 7.58: Distribution of damage eigenvalue D_1 on various x locations

Figure 7.59 shows the distribution of damage eigenvalue D_1 along the x direction at various y locations. Figure 7.59 clearly shows that the damage is largely confined to the region nearest the left-hand end of the beam. The process of damaging the material results in rapid energy dissipation of the normal stress wave and quickly reduces its

energy level below the threshold required to produce further damage. In this sense, the material near the point of load application serves as a buffer to prevent damage to the material located in the beam's interior.

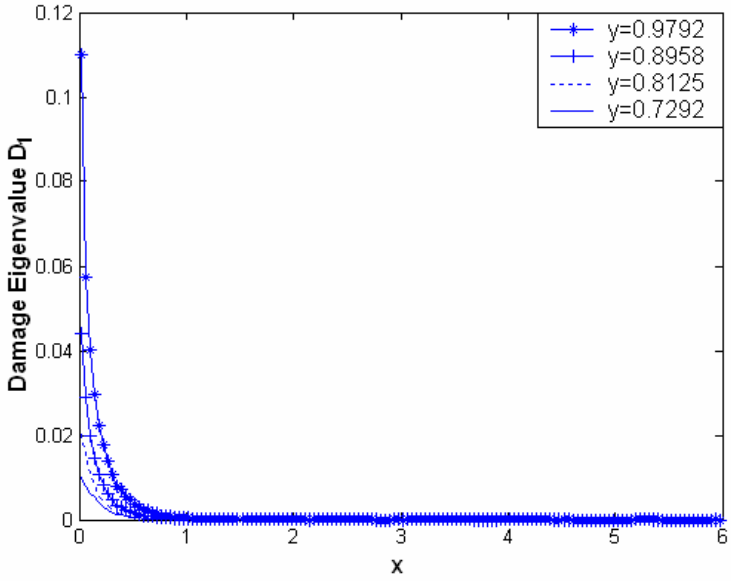


Figure 7.59: Distribution of damage eigenvalue D_1 on various y locations

For comparison, the same problem is solved again using a non-adaptive, uniform refined mesh of 144 elements in the x direction and 72 elements in the y direction (14210 active DOF) with a uniform time step size dt = 0.025/20 = 0.00125 sec. The uniform element size is comparable to the most refined overlay mesh level used previously in the s-adaptive solution and the uniform time step size is also the time step size used previously to start the s-adaptive analysis.

Figure 7.60 shows the comparison of the cumulative distributions of the dominant damage eigenvalue D_1 along the y direction at various x locations by the s-adaptive solution and the uniform refined solution. The two solutions indeed show excellent agreement. After taking into consideration the fact that the s-adaptive solution completes the simulation within 379 time steps using an average number of 3500 active degrees of freedom while the uniform refined solution needs 1200 time steps to finish the same simulation using 14210 active degrees of freedom, it is obvious that the s-adaptive analysis is capable of delivering comparably accurate solution with much less computational cost than the uniform refined analysis, thus resulting in tremendous gain in efficiency.

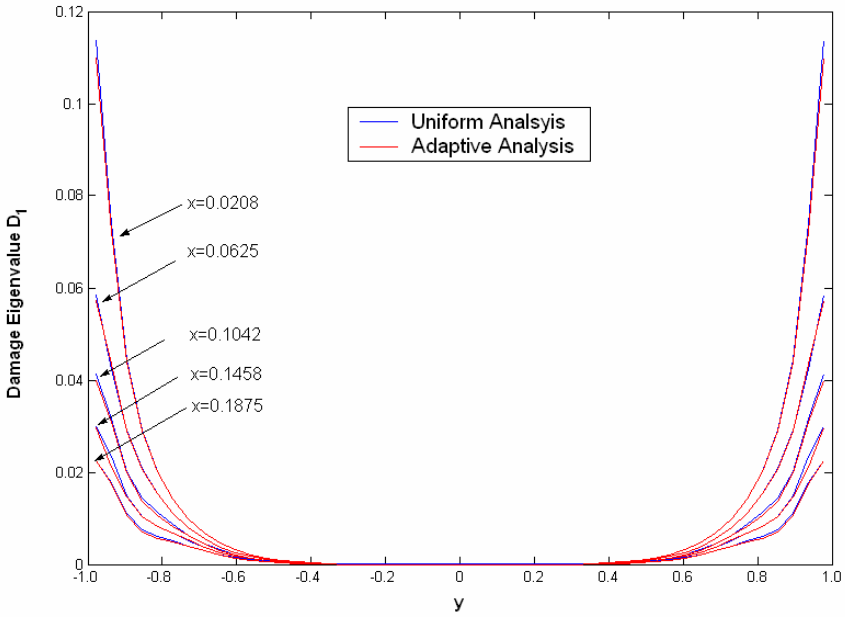


Figure 7.60: Distributions of damage eigenvalue D_1 along the y direction at various x locations

Figures 7.61, 7.62 and 7.63 show the location, size and shape of the stress waves σ_{xx} , σ_{yy} and σ_{xy} predicted by both solutions along different horizontal lines at the discrete times t=0.05, 0.1875, 0.2950 and 0.8900 seconds, respectively. The two solutions show excellent agreement.

Despite the overall excellent agreement between the two solutions, it is also noted in Figure 7.61, 7.62 and 7.63 that the two solutions do not compare very well at the right end of the beam at the early stage of the wave propagation. Starting from t=0 sec., the discrepancies between the two solutions at the right end of the beam propagate leftward into the beam and superpose with the rightward traveling stress waves emanated from the left end of the beam, thus causing discrepancies in the superposed waves. The obvious reason for the discrepancies between the two solutions at the right end of beam is that not enough degrees of freedom were placed at the right end in the s-adaptive analysis, which is confirmed by the composite meshes shown in Figure 7.41 through 7.46. Ironically, it is the discrepancy that embodies the advantage of the adaptive strategy. As shown in Figure 7.39, the right end of beam is only subjected to shear traction forces whose magnitude is far less than the normal traction forces applied on the left end. Naturally, the stress waves generated by the shear traction forces on the right end have much smaller amplitude than the stress waves generated at the left end, which is also confirmed in Figure 7.61 through 7.63. As a result, the high-amplitude stress waves at the left end generate much bigger error than the low-amplitude stress waves at the right end. Recall

that the adaptive procedure allocates the degrees of freedom among the spatial regions which need refinement according to their corresponding contributions to the global error; therefore, it is desirable and inevitable that the degrees of freedom are mainly allocated to the regions with high-amplitude stress waves. If the analyst needs to accurately simulate the stress wave propagation in the regions with low-amplitude stress waves in an automatic adaptive analysis, the only measure is to increase the solution accuracy requirement. In this case, the adaptive analysis tends to become more like a uniform refined analysis because the whole problem domain is being highly refined.

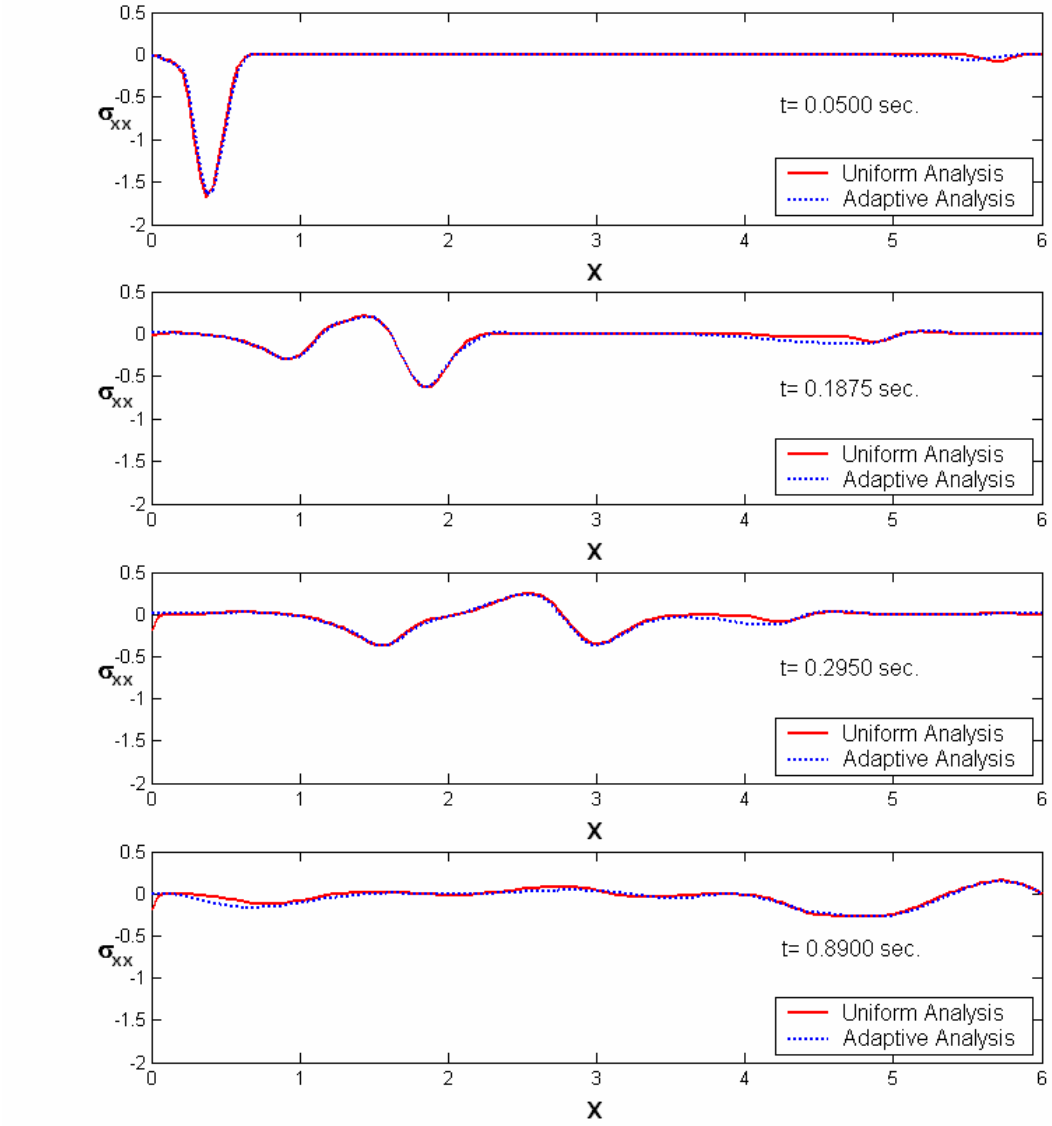


Figure 7.61: Comparison of the computed σ_{xx} distributions along line y=h at various time points by the s-adaptive analysis and the uniform refined analysis

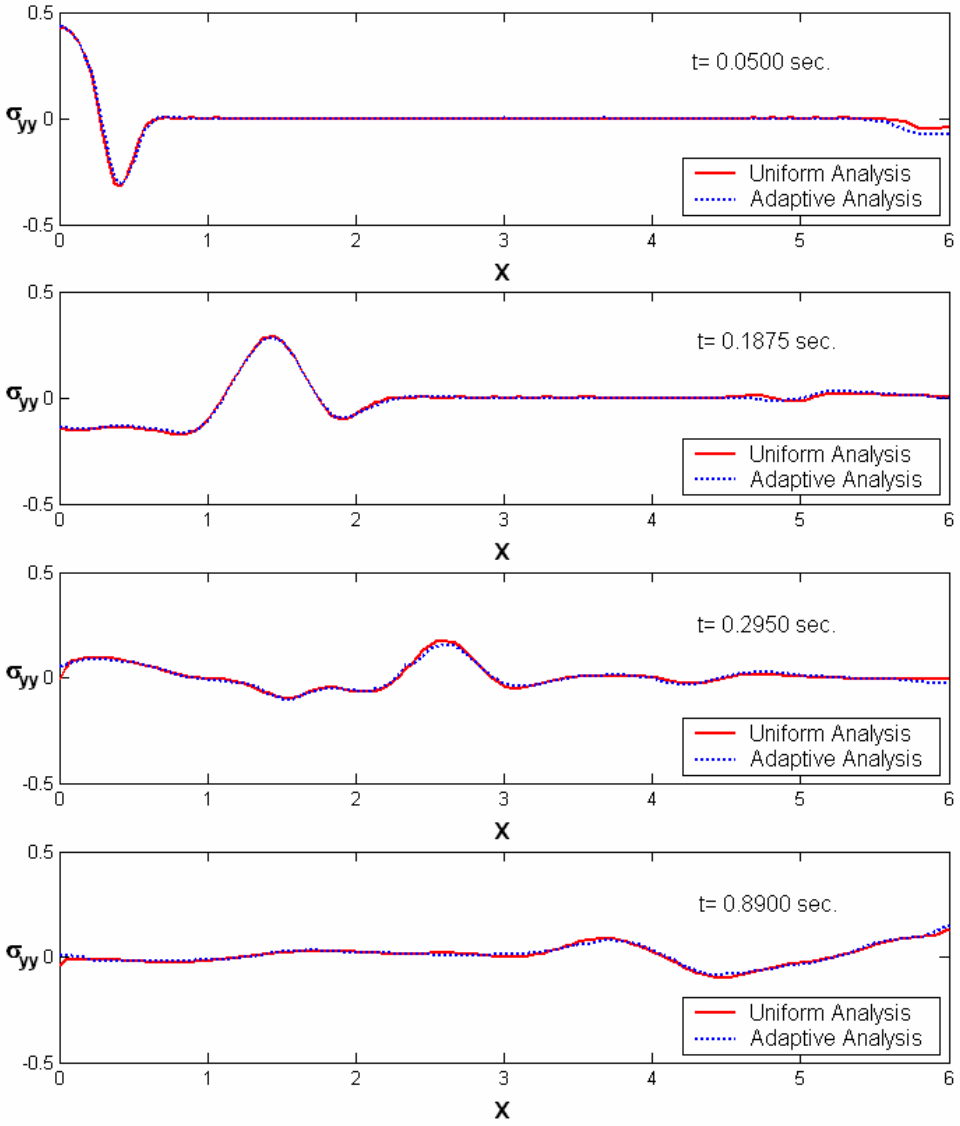


Figure 7.62: Comparison of the computed σ_{yy} distributions along line y=2h/3 at various time points by the s-adaptive analysis and the uniform refined analysis

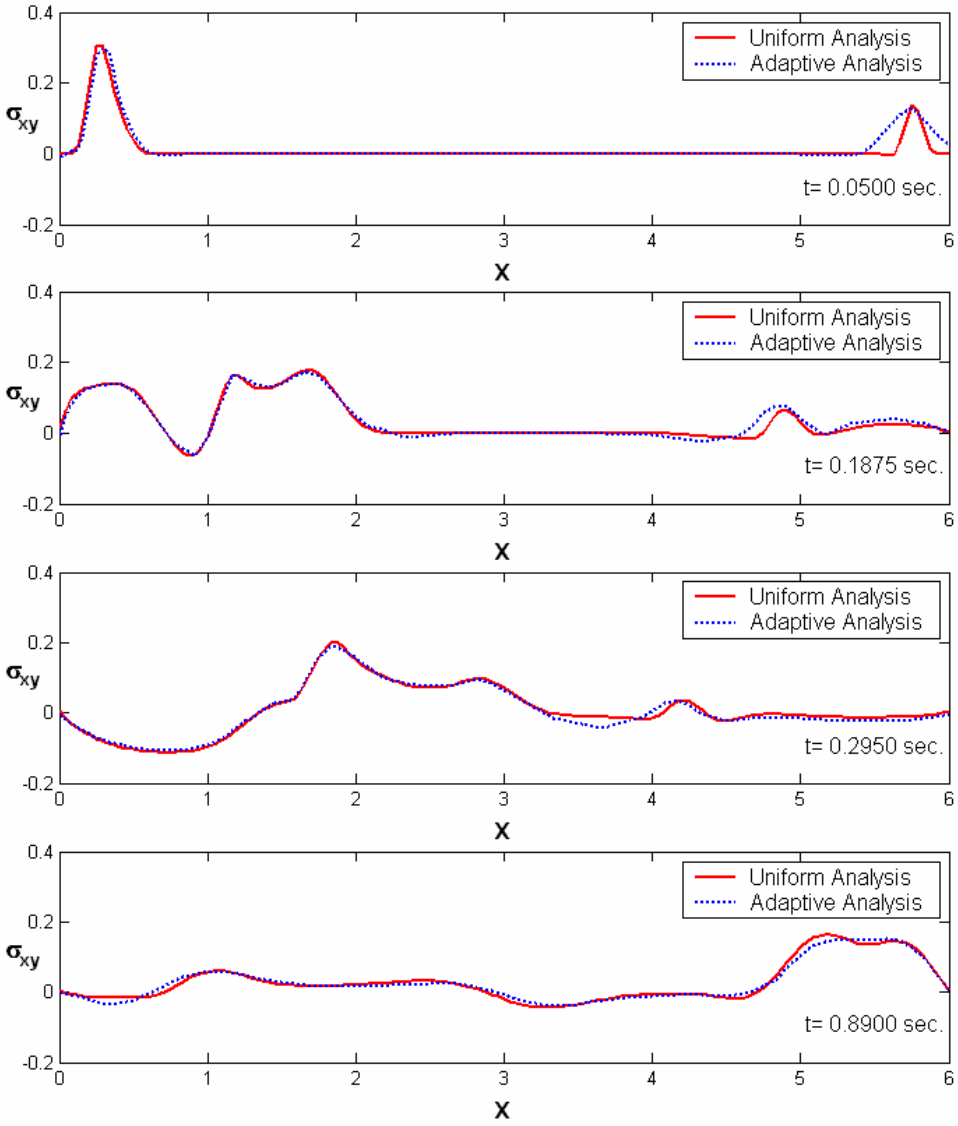


Figure 7.63: Comparison of the computed σ_{xy} distributions along line y=0 at various time points by the s-adaptive analysis and the uniform refined analysis

Chapter 8: CONCLUSIONS AND FUTURE WORK

8.1 Conclusions

The overall goal of this dissertation was to improve the state of the art in adaptive finite element procedures for the transient mechanical response of solids that exhibit linear elastic behavior or material nonlinear behavior caused by progressive damage. The automatic, adaptive, finite element methodology developed in present study simultaneously adjusts both the finite element mesh and the time step size in order to keep the computed spatial and temporal error estimates within user-specified limits over the entire duration of transient analyses. The demonstrated success of the new adaptive method is due in large part to the novel use of structured finite element mesh superposition as the means for providing spatial adaptivity of the mesh, i.e., local mesh refinement and/or local mesh coarsening. This particular type of spatial adaptivity (termed *s-adaptivity*) avoids many of the difficulties and problems associated with the traditional h-adaptive methods that have dominated adaptivity research to date.

While previous researchers have successfully applied s-adaptivity to linear elastic quasi-static problems, the present study represents the first reported extension of the method to transient problems and material nonlinear problems. Due to the fact that transient problems and material nonlinear problems are much more computationally

demanding than linear elastic quasi-static problems, the computational advantages of s-adaptivity prove to be much more significant for transient and/or material nonlinear problems than for quasi-static linear elastic problems.

The advantages afforded by the present structured s-adaptive method are summarized below:

1. Compared to h-adaptive methods that reformulate the entire mesh, the s-adaptive algorithms required for mesh adaptivity (i.e. local mesh coarsening and local mesh refinement) are much simpler to develop and implement, in addition to being much faster to execute.

2. H-adaptive methods that are based on subdivision and consolidation of existing elements inadvertently produce many incompatible (or hanging) nodes which require the introduction of multipoint constraints to maintain spatial continuity of the solution. General algorithms for multipoint constraints are exceedingly difficult to implement for 2-D and 3-D problems. In addition, the presence of multipoint constraints adversely affects the conditioning of the global stiffness matrix, which in turn, adversely affects the convergence rate of iterative solvers. In contrast, the hierarchical data structure of the s-adaptive method completely avoids the need for

cumbersome multipoint constraint algorithms and maintains good stiffness matrix conditioning even in the case of multi-level mesh superposition.

3. In h-adaptive methods that reformulate the entire mesh, the extension from 2-D problems to 3-D problems is quite difficult, due primarily to the complexity of 3-D mesh generation. In contrast, the s-adaptive method does not require a mesh generator, thus the extension of the method from 2-D problems to 3-D problems does not present a disproportionate increase in difficulty.

4. In h-adaptive methods, the process of transferring the current solution from the old mesh to the new mesh introduces significant additional error in the solution. For transient problems, this additional error is particularly troublesome since it represents additional error in the initial conditions that are used to perform the next time step integration. In contrast, the hierarchical data structure of the s-adaptive method largely avoids such solution-transfer errors. More specifically, for local mesh refinement, the s-adaptive method does not introduce any additional error into the solution, while for local mesh coarsening, the s-adaptive method introduces a minimal amount of additional error into the solution (loss of some high-frequency, low-amplitude resolution).

Extension of the s-adaptive methodology to linear transient problems and material nonlinear transient problems presented a number of challenging issues that had to be resolved in the course of the present research. Many of these issues are directly related to the need for increased execution speed of the spatial adaptivity algorithm. In quasi-static, linear elastic problems, only one sequence of adaptive meshes is required during the adaptive solution process and therefore, the speed of the mesh adaptivity algorithm is not a critical factor. However, transient problems and nonlinear problems involve obtaining linearized solutions at many different time steps or load steps, each of which requires a sequence of adaptive meshes. In this case, the execution speed of the adaptive algorithm is of paramount importance. In addition, transient problems require that the solution must be passed from one adaptive mesh to the next adaptive mesh with a bare minimum of solution-transfer error since this form of error compromises the initial conditions used for the next time integration step. Thus, the choices made in developing the present s-adaptive algorithm were largely influenced by the need for maximum execution speed and minimum solution-transfer error.

In the present dissertation, the performance characteristics of the s-adaptive method were examined for a number of different solid mechanics problem types including 2-D and 3-D linear elastic quasi-static problems, 2-D material nonlinear quasi-static problems, and 2-D transient problems for linear elastic and material nonlinear materials. In each case, s-adaptive solutions are compared to solutions obtained using non-adaptive, uniformly

refined meshes. In each case, the element size of the most refined non-adaptive uniformly refined mesh was chosen to be the same as the smallest element that was generated during the s-adaptive solution. These comparisons demonstrated that the s-adaptive method is capable of generating solutions with the same accuracy level as a non-adaptive, uniform refined mesh; however, the s-adaptive solution uses far fewer DOF and consequently executes much faster and uses less computer memory.

The novel contributions of the present work are summarized below.

To minimize the computational effort required in estimating the spatial error, the widely accepted Superconvergent Patch Recovery (SPR) procedure of Zienkiewicz and Zhu was adapted to multi-level composite meshes, resulting in a *Layer-by-Layer SPR* scheme. In addition, the robustness of the basic SPR procedure was increased by developing a *rotated SPR* procedure which avoids the rank deficiency problem that can be triggered by certain geometric patterns of element patches.

As part of the overall research effort, the basic finite element mesh superposition scheme was extended from 2-D linear elastic quasi-static analysis (as previously published in literature) to the more complex cases of: a) 3-D linear elastic quasi-static analysis, b) 2-D nonlinear quasi-static analysis, c) 2-D linear elastic transient analysis, and d) and 2-D nonlinear transient analysis. These extensions are the first of their kinds to be reported.

In particular, the extension from 2-D problems to 3-D problems was shown to be relatively straightforward, and did not require a disproportionate increase in algorithm development difficulty or solution time. The extension of s-adaptivity to transient problems required the development of effective strategies for coordinating the temporal adaptivity and spatial adaptivity. In the extension of s-adaptivity to material nonlinear problems, the material nonlinear problems lead to spatially varying material properties which necessitate the development of new methods of efficiently computing the system matrices and system vectors for the composite system of equations. Furthermore, the extension of the s-adaptive method to material nonlinear problems resulted in the development of the *mastergrid* data transfer scheme to ensure that the computed distribution of internal (or element-wise) variables is accurately transferred between consecutive composite meshes.

8.2 Future Work

Adaptive finite element analysis remains an active research area which is far from mature. Based on the results of this study, s-adaptive methods are concluded to be a viable adaptive methodology with many advantages and worthy of further research efforts. In the context of the proposed s-adaptive procedure, based on the experiences and thoughts obtained in the course of the study, we believe that the following issues are to investigate:

Avoiding unnecessary decrease of time step size after mesh changes.

In the adaptive analyses in this research, after each mesh change, the displacement and velocity fields are transferred from the old composite mesh to the new composite mesh. In contrast, the acceleration field and the third order derivative of the displacement, which is needed in temporal error estimation, are directly calculated by resolving the dynamic equations on the new composite mesh [45]. Somehow in the HHT-α time integration method adopted in this study, this approach sometimes causes dramatic increase of the temporal error immediately after mesh changes and consequently, the dramatic decrease of the time step size. In most cases, this dramatic decrease of time step size is suspected to be unnecessary because the time step size is usually observed to increase gradually in the subsequent time steps. This over-reduction in the time step size that occurs immediately after mesh changes is most likely due to inadequacy in the temporal error estimate adopted in this study. Therefore, the development of new temporal error estimates should be pursued to avoid this issue.

Expansion of the element library.

In this study, the 4-node 2-D quadrilateral element is used for all 2-D problems. Other types of commonly used 2-D elements, such as the 3-node triangle element, the 8-node quadrilateral element, and plate and shell elements, should be investigated to determine their performance characteristics within the context of s-adaptive methods.

Extension to dynamic analysis of the three-dimensional problems.

To date, most of the studies on adaptive finite element methods have focused on 2-D problems. In comparison, 3-D adaptive studies have received very little attention, probably due to the fact that most of the 2-D adaptive methods are difficult and cumbersome to extend to 3-D problems. In the present study, it was demonstrated that the s-adaptive method can be extended form 2-D to 3-D without a disproportionate increase in difficulty. However, the present study only demonstrated this extension for the quasi-static case. Future efforts should investigate the performance characteristics of the s-adaptive method problems for 3-D transient problems where it is anticipated that the unique features of the s-adaptive method will be particularly advantageous.

Extension to other types of nonlinear dynamic problems.

In this study, the s-adaptive method was shown to be effective for material nonlinear problems where the material exhibits a softening behavior caused by progressive damage. In many ways, this type of material nonlinearity is quite similar to plasticity. Future studies should further investigate the application of the s-adaptive method to a broader range of nonlinear behaviors.

Appendix

Appendix A: The Implicit Nonlinear Finite Element Method Using HHT-α Time Integration

The HHT-α variant [39] of the Newmark direct integration method consists of the recursive equations:

$$\mathbf{M}\ddot{\mathbf{d}}_{n+1} + (1+\alpha)\mathbf{C}\dot{\mathbf{d}}_{n+1} - \alpha\mathbf{C}\dot{\mathbf{d}}_{n} + (1+\alpha)\mathbf{K}\mathbf{d}_{n+1} - \alpha\mathbf{K}\mathbf{d}_{n} = \mathbf{F}(t_{n+1+\alpha}) \tag{1a}$$

$$\dot{\mathbf{d}}_{n+1} = \dot{\mathbf{d}}_{n} + [(1-\gamma)\ddot{\mathbf{d}}_{n} + \gamma\ddot{\mathbf{d}}_{n+1}]\Delta t_{n} \tag{1b}$$

$$\mathbf{d}_{n+1} = \mathbf{d}_{n} + \dot{\mathbf{d}}_{n}\Delta t_{n} + [(1-2\beta)\ddot{\mathbf{d}}_{n} + 2\beta\ddot{\mathbf{d}}_{n+1}]\frac{\Delta t_{n}^{2}}{2} \tag{1c}$$

The acceleration in equation (1c) can be expressed as

$$\ddot{\mathbf{d}}_{n+1} = \frac{1}{\beta\Delta t_{n}^{2}}[(\mathbf{d}_{n+1} - \mathbf{d}_{n}) - \dot{\mathbf{d}}_{n}\Delta t_{n} - (\frac{1}{2\beta} - 1)\ddot{\mathbf{d}}_{n}] \tag{1d}$$

In the implicit nonlinear finite element method, equation (1a) is required to be satisfied any time and the state variables are known at time t_n in the time step $[t_n, t_{n+1}]$ for the solution of the state variables at t_{n+1}. Therefore, the solution procedure aims to find the state variables that satisfy

$$\mathbf{M}\ddot{\mathbf{d}}_{n+1}^{i+1} + (1+\alpha)\mathbf{C}\dot{\mathbf{d}}_{n+1}^{i+1} - \alpha\mathbf{C}\dot{\mathbf{d}}_{n} + (1+\alpha)\mathbf{K}_{n+1}^{i+1}\mathbf{d}_{n+1}^{i+1} - \alpha\mathbf{K}\mathbf{d}_{n} = \mathbf{F}(t_{n+1+\alpha}) \tag{2}$$

where the superscript *i* denotes the number of iterations accounting for the nonlinearity.

Without the loss of generality, it is safe to assume $\alpha = 0$ in equation (2) to simplify the derivation.

$$\mathbf{M}\ddot{\mathbf{d}}_{\mathrm{n+1}}^{i+1} + \mathbf{C}\dot{\mathbf{d}}_{\mathrm{n+1}}^{i+1} + \mathbf{K}_{\mathrm{n+1}}^{i+1}\mathbf{d}_{\mathrm{n+1}}^{i+1} = \mathbf{F}(t_{\mathrm{n+1}}) \tag{3}$$

In order to linearize equation (3), we express $\mathbf{d}_{\mathrm{n+1}}^{i+1}$ as follows

$$\mathbf{d}_{\mathrm{n+1}}^{i+1} = \mathbf{d}_{\mathrm{n+1}}^{i} + \Delta\mathbf{d}_{\mathrm{n+1}}^{i} \tag{4}$$

Linearization of the third term in equation (3) about $\mathbf{d}_{\mathrm{n+1}}^{i}$ yields

$$\mathbf{K}_{\mathrm{n+1}}^{i+1}\mathbf{d}_{\mathrm{n+1}}^{i+1} \approx \mathbf{K}_{\mathrm{n+1}}^{i}\left(\mathbf{d}_{\mathrm{n+1}}^{i} + \Delta\mathbf{d}_{\mathrm{n+1}}^{i}\right) = \mathbf{K}_{\mathrm{n+1}}^{i}\mathbf{d}_{\mathrm{n+1}}^{i} + \mathbf{K}_{\mathrm{n+1}}^{i}\Delta\mathbf{d}_{\mathrm{n+1}}^{i} \tag{5}$$

where we define the internal force vector at the *i*th iteration of time t_{n+1} as

$$\mathbf{K}_{\mathrm{n+1}}^{i}\mathbf{d}_{\mathrm{n+1}}^{i} = \mathbf{F}^{\mathrm{INT}}\left(\mathbf{d}_{\mathrm{n+1}}^{i}\right) = \mathbf{I}_{t_{\mathrm{n+1}}}^{i} \tag{6}$$

Plugging the resulting equations (5) and (6) into equation (3) yields

$$\mathbf{M}\ddot{\mathbf{d}}_{\mathrm{n+1}}^{i+1} + \mathbf{C}\dot{\mathbf{d}}_{\mathrm{n+1}}^{i+1} + \mathbf{K}_{\mathrm{n+1}}^{i}\Delta\mathbf{d}_{\mathrm{n+1}}^{i} = \mathbf{F}(t_{\mathrm{n+1}}) - \mathbf{I}_{t_{\mathrm{n+1}}}^{i} \tag{7}$$

Equation (7) can be further decomposed using equation (1b) to obtain

$$\mathbf{M}\ddot{\mathbf{d}}_{\mathrm{n+1}}^{i+1} + \left[\mathbf{C}\dot{\mathbf{d}}_{\mathrm{n}} + (1-\gamma)\Delta t_{\mathrm{n}}\mathbf{C}\ddot{\mathbf{d}}_{\mathrm{n}} + \gamma\Delta t_{\mathrm{n}}\mathbf{C}\ddot{\mathbf{d}}_{\mathrm{n+1}}^{i+1}\right] + \mathbf{K}_{\mathrm{n+1}}^{i}\Delta\mathbf{d}_{\mathrm{n+1}}^{i} = \mathbf{F}(t_{\mathrm{n+1}}) - \mathbf{I}_{t_{\mathrm{n+1}}}^{i} \tag{8a}$$

Grouping the coefficients of $\ddot{\mathbf{d}}_{\mathrm{n+1}}^{i+1}$ yields

$$\left(\mathbf{M} + \gamma\Delta t_{\mathrm{n}}\mathbf{C}\right)\ddot{\mathbf{d}}_{\mathrm{n+1}}^{i+1} + \mathbf{C}\dot{\mathbf{d}}_{\mathrm{n}} + (1-\gamma)\Delta t_{\mathrm{n}}\mathbf{C}\ddot{\mathbf{d}}_{\mathrm{n}} + \mathbf{K}_{\mathrm{n+1}}^{i}\Delta\mathbf{d}_{\mathrm{n+1}}^{i} = \mathbf{F}(t_{\mathrm{n+1}}) - \mathbf{I}_{t_{\mathrm{n+1}}}^{i} \tag{8b}$$

Using equation (1d) and further equation (4) in equation (8b) yields

$$\begin{aligned}&\left(\mathbf{M}+\gamma\Delta t_{\mathrm{n}}\mathbf{C}\right)\frac{1}{\beta\Delta t_{\mathrm{n}}{}^{2}}[(\mathbf{d}_{\mathrm{n+1}}^{i+1}-\mathbf{d}_{\mathrm{n}})-\dot{\mathbf{d}}_{\mathrm{n}}\Delta t_{\mathrm{n}}-(\frac{1}{2\beta}-1)\ddot{\mathbf{d}}_{\mathrm{n}}]\\&+\mathbf{C}\dot{\mathbf{d}}_{\mathrm{n}}+(1-\gamma)\Delta t_{\mathrm{n}}\mathbf{C}\ddot{\mathbf{d}}_{\mathrm{n}}+\mathbf{K}_{\mathrm{n+1}}^{i}\Delta\mathbf{d}_{\mathrm{n+1}}^{i}=\mathbf{F}(t_{\mathrm{n+1}})-\mathbf{I}_{t_{\mathrm{n+1}}}^{i}\end{aligned}\tag{9a}$$

$$\begin{aligned}&\left(\mathbf{M}+\gamma\Delta t_{\mathrm{n}}\mathbf{C}\right)\frac{1}{\beta\Delta t_{\mathrm{n}}{}^{2}}[(\mathbf{d}_{\mathrm{n+1}}^{i}+\Delta\mathbf{d}_{\mathrm{n+1}}^{i}-\mathbf{d}_{\mathrm{n}})-\dot{\mathbf{d}}_{\mathrm{n}}\Delta t_{\mathrm{n}}-(\frac{1}{2\beta}-1)\ddot{\mathbf{d}}_{\mathrm{n}}]\\&+\mathbf{C}\dot{\mathbf{d}}_{\mathrm{n}}+(1-\gamma)\Delta t_{\mathrm{n}}\mathbf{C}\ddot{\mathbf{d}}_{\mathrm{n}}+\mathbf{K}_{\mathrm{n+1}}^{i}\Delta\mathbf{d}_{\mathrm{n+1}}^{i}=\mathbf{F}(t_{\mathrm{n+1}})-\mathbf{I}_{t_{\mathrm{n+1}}}^{i}\end{aligned}\tag{9b}$$

Finally, the system of linear equations for implicit nonlinear dynamics is given as

$$\begin{aligned}&\left(\frac{\mathbf{M}+\gamma\Delta t_{\mathrm{n}}\mathbf{C}}{\beta\Delta t_{\mathrm{n}}{}^{2}}+\mathbf{K}_{\mathrm{n+1}}^{i}\right)\Delta\mathbf{d}_{\mathrm{n+1}}^{i}=\mathbf{F}(t_{\mathrm{n+1}})-\mathbf{I}_{t_{\mathrm{n+1}}}^{i}+\left[\frac{\mathbf{M}+\gamma\Delta t_{\mathrm{n}}\mathbf{C}}{\beta\Delta t_{\mathrm{n}}}-\mathbf{C}\right]\dot{\mathbf{d}}_{\mathrm{n}}+\\&\left[(\frac{1}{2\beta}-1)\left(\mathbf{M}+\gamma\Delta t_{\mathrm{n}}\mathbf{C}\right)-(1-\gamma)\Delta t_{\mathrm{n}}\mathbf{C}\right]\ddot{\mathbf{d}}_{\mathrm{n}}-\frac{\mathbf{M}+\gamma\Delta t_{\mathrm{n}}\mathbf{C}}{\beta\Delta t_{\mathrm{n}}{}^{2}}\mathbf{d}_{\mathrm{n+1}}^{i}+\frac{\mathbf{M}+\gamma\Delta t_{\mathrm{n}}\mathbf{C}}{\beta\Delta t_{\mathrm{n}}{}^{2}}\mathbf{d}_{\mathrm{n}}\end{aligned}\tag{10}$$

After the displacement increment $\Delta\mathbf{d}_{\mathrm{n+1}}^{i}$ is solved in equation (10), the displacement, velocity and acceleration at *(i+1)*th iteration of time t_{n+1} can be obtained by using equations (4,1b,1d), respectively.

References

1. Clough, R.W. *The finite element method in plane stress analysis*. in *Proc. 2nd ASCE Conf. On Electronic Computation*. 1960. Pittsburgh,PA.

2. Babuska, I. *Courant element: before and after*. in *Finite Element Methods, fifty years of the Courant element*. 1993.

3. Zienkiewicz, O.Z., *Origin, milestones and directions of the finite element method-a personal view.* Archives of Computational Methods in Engineering, 1995. **2**(1): p. 1-48.

4. Babuska, I. and W.C. Rheinboldt, *A posteriori error estimates for the finite element method.* International Journal for Numerical Methods in Engineering, 1978. **11**: p. 1597-1615.

5. Zienkiewicz, O.Z. and R.L. Taylor, *The finite element method, basic formulation and linear problems*. 4 ed. Vol. 2. 1991, New York: McGraw-Hill.

6. Zienkiewicz, O.Z. and J.Z. Zhu, *A simple error estimator and adaptive procedure for practical engineering analysis.* International Journal for Numerical Methods in Engineering, 1987. **24**: p. 333-357.

7. Zienkiewicz, O.Z. and J.Z. Zhu, *The superconverent patch recovery and a posteriori error estimates, Part Ⅰ: the recovery technique.* International Journal for Numerical Methods in Engineering, 1992. **33**: p. 1331-1364.

8. Zienkiewicz, O.Z. and J.Z. Zhu, *The superconverent patch recovery and a posteriori error estimates, Part Ⅱ: error estimates and adaptivity.* International Journal for Numerical Methods in Engineering, 1992. **33**: p. 1365-1382.

9. Zienkiewicz, O.Z., B. Boroomand, and J.Z. Zhu, *Recovery procedures in error estimation and adaptivity Part I: Adaptivity in linear problems.* Comput. Methods Appl. Mech. Engrg., 1999. **176**: p. 111-125.

10. Wiberg, N.E. and F. Abdulwahab, *Patch recovery based on superconvergent derivatives and equilibrium.* International Journal for Numerical Methods in Engineering, 1993. **36**: p. 2703-2724.

11. Wiberg, N.E., F. Abdulwahab, and S. Ziukas, *Enhanced superconvergent patch recovery incorporating equilibrium and boundary conditions.* International Journal for Numerical Methods in Engineering, 1994. **37**: p. 3417-3440.

12. Wiberg, N.E., F. Abdulwahab, and S. Ziukas, *Improved element stresses for node and element patches using superconvergent patch recovery.* Commun. Numer. Meth. Engng., 1995. **11**: p. 619-627.

13. Li, L.Y. and P. Bettess, *Adaptive finite element methods: A review.* Applied mechanics review, 1997. **50**(10): p. 581-591.

14. Choi, C.K. and N.H. Lee, *A 3-D adaptive mesh refinement using variable-node solid transition elements.* International Journal for Numerical Methods in Engineering, 1996. **39**(9): p. 1585-1606.

15. Fish, J., *The s-version of the finite element method.* Computers and Structures, 1990. **43**: p. 593-547.

16. Fish, J. and R. Guttal, *The s-version of the finite element method for laminated composites.* International Journal for Numerical Methods in Engineering, 1996. **39**: p. 3641-3662.

17. Fish, J. and S. Markolefas, *The s-version of the finite element method for multilayer laminates.* International Journal for Numerical Methods in Engineering, 1992. **33**: p. 1081-1105.

18. Reddy, J.N. and D.H. Robbins, Jr., *Theories and computational models for composite laminates.* Applied Mechanics Reviews, 1994. **47**(6): p. 147-169.

19. Robbins, D.H., Jr., *Hierarchical modeling of laminated composite plates using variable kinematic finite elements and mesh superposition*, in *Department of Engineering Science and Mechanics*. 1993, Virginia Polytechnic Institute and State University: Blacksburg, VA.

20. Robbins, D.H., Jr. and J.N. Reddy, *Variable kenematic modeling of laminated composite plates.* International Journal for Numerical Methods in Engineering, 1996. **39**: p. 2283-2317.

21. Takano, N., M. Zako, and M. Ishizono, *Multi-scale computational method for elastic bodies with global and local heterogeneity.* J. Computer-Aided Material Design, 2000. **7**: p. 111-132.

22. Darbha, K. and A. Dasgupta, *A nested finite element methodology (NFEM) for stress analysis of electronic products—Part Ⅰ: theory and formulation.* Journal of Electronic Packaging, 2001. **123**: p. 141-146.

23. Darbha, K. and A. Dasgupta, *A nested finite element methodology (NFEM) for stress analysis of electronic products—Part Ⅱ: durability analysis of flip chip and chip scale interconnects.* Journal of Electronic Packaging, 2001. **123**: p. 147-155.

24. Fish, J. and S. Markolefas, *Adaptive s-method for linear elastostatics.* Comput. Methods Appl. Mech. Engrg., 1993. **104**: p. 363-396.

25. Fish, J., et al., *On adaptive multilevel superposition of finite element meshes for linear elastostatics.* Applied Numerical Mathematics, 1994. **14**(135-164).

26. Park, J.W., J.W. Hwang, and Y.H. Kim, *Efficient finite element analysis using mesh superposition technique.* Finite elements in analysis and design, 2003. **39**: p. 619-638.

27. Babuska, I., et al., *Accuracy Estimates and Adaptive Refinement in Finite Element Computations*. 1986, New York: Wiley.

28. Demkowicz, L., E.R.d.A. Oliveira, and I. Babuska, *Proceedings of the second workshop on reliability in computational mechanics.* Comput. Methods Appl. Mech. Engng., 1992. **101**: p. 1-491.

29. Oden, J.T., *Proceedings of the workshop on reliability in computational mechanics.* Comput. Methods Appl. Mech. Engng., 1990. **82**: p. 1-388.

30. Bathe, K.J., *Finite element procedures in engineering analysis*. 1982, Englewood Cliffs, NJ: Prentice-Hall.

31. Hughes, T.J.R., *The Finite Element Method: Linear Static and Dynamic Finite Element Analysis*. 1987, Englewood, NJ: Prentice-Hall Inc.

32. Zienkiewicz, O.Z. and R.L. Taylor, *The finite element method, solid and fluid mechanics, dynamics and non-linearity*. 1991, New York: McGraw-Hill.

33. Joo, K.J. and E.L. Wilson, *An adaptive finite element technique for structural dynamic analysis.* Computers and Structures, 1988. **30**: p. 1319-1339.

34. Cook, R.D. and J. Avrashi, *Error estimation and adaptive meshing for vibration problems.* Computers and Structures, 1993. **44**: p. 207-213.

35. Belyschko, T. and M. Tabbara, *H-adaptivity finite element methods for dynamic problems, with emphasis on localization.* International Journal for Numerical Methods in Engineering, 1993. **36**: p. 4245-4265.

36. Wiberg, N.E., L.F. Zeng, and X.D. Li, *Error estimation and adaptivity in elastodynamics.* Comput. Methods Appl. Mech. Engng., 1992. **101**: p. 369-395.

37. Zeng, L.F. and N.E. Wiberg, *Spatial mesh adaptation in semidiscrete finite element analysis of linear elastodynamic problems.* Comput. Mech., 1992. **9**: p. 315-332.

38. Newmark, N.M., *A method of computation for structural dynamics.* J. Eng. Mech. Div. ASCE, 1959. **85**(EM3): p. 67-94.

39. Hilber, H.M., T.J.R. Hughes, and R.L. Taylor, *Improved numerical dissipation for time integration algorithms in structural dynamics.* Earthquake Eng. Struct. Dyn., 1977. **5**: p. 283-292.

40. Bergan, P.G. and E. Mollestad, *An automatic time-stepping algorithm for dynamic problems.* Comput. Methods Appl. Mech. Engng., 1985. **49**: p. 299-318.

41. Zienkiewicz, O.Z. and Y.M. Xie, *A simple error estimator and adaptive time stepping procedure for dynamic analysis.* Earthquake Eng. Struct. Dyn., 1991. **20**: p. 871-887.

42. Zeng, L.F., et al., *A posteriori error estimation and adaptive time-stepping for Newmark integration in dynamic analysis.* Earthquake Eng. Struct. Dyn., 1992. **21**: p. 555-571.

43. Wiberg, N.E. and X.D. Li, *A post-processing technique and a posteriori error estimate for the newmark method in dynamic analysis.* Earthquake Eng. Struct. Dyn., 1993. **22**: p. 465-489.

44. Choi, C.K. and H.J. Chung, *Error estimates and adaptive time stepping for various direct time integration methods.* Computers and Structures, 1996. **60**(6): p. 923-944.

45. Wiberg, N.E. and X.D. Li, *A postprocessed error estimate and an adaptive procedure for the semidiscrete finite element method in dynamic analysis.* International Journal for Numerical Methods in Engineering, 1994. **37**: p. 3585-3603.

46. Choi, C.K. and H.J. Chung, *An adaptive control of spatial-temporal discretization error in finite element analysis of dynamic problems.* Struct. Engng Mech., 1995. **3**: p. 391-410.

47. Belyschko, T., B.L. Wong, and E.J. Plaskacz, *Fission-fusion adaptivity in finite element analysis for nonlinear dynamics of shells.* Computers and Structures, 1988. **33**(5): p. 1307-1323.

48. Boroomand, B. and O.Z. Zienkiewicz, *Recovery procedures in error estimation and adaptivity. Part Ⅱ: Adaptivity in nonlinear problems of elasto-plasticity behaviour.* Comput. Methods Appl. Mech. Engrg., 1999. **176**: p. 127-146.

49. Ortiz, M. and J.J. Quigley, *Adaptive mesh refinement in strain localization problems.* Comput. Methods Appl. Mech. Engrg., 1991. **90**: p. 781-804.

50. Johnson, C. and P. Hansbo, *Adaptive finite element methods in computational mechanics.* Comput. Methods Appl. Mech. Engrg., 1992. **101**: p. 143-181.

51. Li, L.Y. and P. Bettess, *Error estimates and adaptive remeshing techniques in elasto-plasticity.* Commun. Numer. Meth. Engng., 1997. **13**: p. 285-299.

52. Kato, K., N.S. Lee, and K.J. Bathe, *Adaptive finite element analysis of large strain elastic response.* Computers and Structures, 1993. **47**: p. 829-855.

53. Sandhu, J.S. and H. Liebowitz, *Examples of adaptive FEA in plasticity.* Engineering Fracture Mechanics, 1995. **50**: p. 947-956.

54. Wiberg, N.E. and X.D. Li, *Adaptive finite element procedures for linear and non-linear dynamics.* International Journal for Numerical Methods in Engineering, 1999. **46**: p. 1781-1802.

55. Peric, D., J. Yu, and D.R.J. Owen, *On error estimates and adaptivity in elasto-plastic solids: Application to the numerical simulation of strain localization in classical and Cosserat continua.* International Journal for Numerical Methods in Engineering, 1994. **37**: p. 1351-1379.

56. Gallimard, L., P. Ladeveze, and J.P. Pelle, *Error estimation and adaptivity in elastoplasticity.* International Journal for Numerical Methods in Engineering, 1996. **39**: p. 189-217.

57. Peric, D., et al., *Transfer operators for evolving meshes in small strain elasto-plasticity.* Comput. Methods Appl. Mech. Engrg., 1996. **137**: p. 331-344.

58. Camacho, G.T. and M. Ortiz, *Adaptive Lagrangian modeling of ballistic penetration of metallic targets.* Comput. Methods Appl. Mech. Engrg., 1997. **142**: p. 269-301.

59. Lee, N.S. and K.J. Bathe, *Error indicators and adaptive remeshing in large deformation finite element analysis.* Finite Elem. Anal. Des., 1994. **16**: p. 99-139.

60. Zienkiewicz, O.Z. and B. Boroomand, *Adaptive mesh update in plasticity problems and an improved solution process using SPR, A preliminary study*. Advances in Finite Element Technology, ed. N.E. Wiberg. 1995. 1-21.

61. Molinari, J.F. and M. Ortiz, *Three-dimensional adaptive meshing by subdivision and edge-collapse in finite-deformation dynamic-plasticity problems with application to adiabatic shear banding.* International Journal for Numerical Methods in Engineering, 2002. **53**: p. 1101-1126.

62. Belyschko, T. and I.S. Yeh, *Adaptivity in nonlinear structural dynamics with contact-impact*. Adaptive, Multilevel and Hierarchical Computational Strategies, ed. A.K. Noor. Vol. 157. 1992: ASME. 165-202.

63. Bruce, E.E. and G.W. Robert, *Large spatial, temporal and algorithmic adaptivity for implicit nonlinear finite element analysis*. Adaptive, Multilevel and Hierarchical Computational Strategies, ed. A.K. Noor. Vol. 157. 1992: ASME. 203-222.

64. Zienkiewicz, O.Z., Y.C. Liu, and G.C. Huang, *Error estimation and adaptivity in flow formulation for forming problems.* International Journal for Numerical Methods in Engineering, 1988. **25**: p. 23-42.

65. Yue, Z. and D.H. Robbins, Jr., *Adaptive superposition of finite element meshes in elastodynamic problems.* International Journal for Numerical Methods in Engineering, 2005. **63**(11): p. 1604-1635.

66. Yue, Z. and D.H. Robbins, Jr., *Rank deficiency in superconvergent patch recovery techniques with 4-node quadrilateral elements.* Commun. Numer. Meth. Engng., (submitted in May 2005).

67. Oden, J.T. and J.N. Reddy, *An introduction to the mathematical theory of finite elements*. 1976, New York: Wiley.

68. Zienkiewicz, O.Z. and Y.K. Cheung, *Finite Element Method in Structural Mechanics*. 1976: McGraw-Hill.

69. Hinton, E. and J. Campbell, *Local and global smoothing of discontinuous function using a least squares method.* International Journal for Numerical Methods in Engineering, 1974. **8**: p. 461-480.

70. Wiberg, N.E., *Superconvergent Patch Recovery—a key to quality assessed FE solutions.* advances in engineering software, 1997. **28**: p. 85-95.

71. Kim, Y.M., I. Levit, and G. Stanley. *A finite element adaptive mesh refinement technique that avoids multipoint constraints and transition zones*. in *Iterative Equation Solvers for Structural Mechanics Problems*. 1991. Parsons, ID: ASME CED.

72. Saad, Y., *Iterative Methods for sparse linear systems*. 1996, Boston: PWS Pub. Co.

73. Jouglard, C.E. and A.L.G.A. Coutinho, *A comparison of iterative multi-level finite element solvers.* Computers and Structures, 1998. **69**: p. 655-670.

74. Zienkiewicz, O.Z. and J.Z. Zhu, *The superconverent patch recovery (SPR) and adaptive finite element refinement.* Comput. Methods Appl. Mech. Engrg., 1992. **101**: p. 207-224.

75. Blacker, T. and T. Belyschko, *Superconvergence patch recovery with equilibrium and conjoint interpolation enhancements.* International Journal for Numerical Methods in Engineering, 1994. **37**: p. 517-536.

76. Labbe, P. and A. Garon, *A robust implementation of Zienkiewicz and Zhu's local patch recovery method.* Commun. Numer. Meth. Engng., 1995. **11**: p. 427-434.

77. Zienkiewicz, O.Z., J.Z. Zhu, and J. Wu, *Superconvergent Recovery Techniques - some further tests.* Commun. Numer. Meth. Engng., 1993. **9**: p. 251-258.

78. Taylor, R.L., *FEAP: A Finite Element Analysis Program*. 1999, http://www.ce.berkeley.edu/~rlt/feap/.

79. Kachanov, L.M., *On the Creep Fracture Time.* Izv Akad. Nauk USSR Otd. Tekh., 1958. **8**: p. 26-31.

80. Reddy, J.N. and D.H. Robbins, Jr., *Computational Modeling of Damage and Failures in Composite Laminates*. Encyclopedia of Computational Mechanics, Volume 2: Solid and Structures. 2004, New York: John Wiley & Sons.

81. Lemaitre, J., *A Course on Damage Mechanics*. 2nd ed. 1996, Berlin: Springer.

82. Skrzypek, J. and A. Ganczarski, *Modeling of Material Damage and Failure of Structures*. 1999, Berlin: Springer.

83. Kolsky, H. *An investigation of mechanical properties of materials at very high rates of loading*. in *Proceedings of the Physics Society of London*. 1949.